The Generative Learning and Discriminative Fitting of Linear Deformable Models

Jason Mora Saragih

A thesis submitted for the degree of Doctor of Philosophy of
The Australian National University

1 October 2008

Research School of Information Sciences and Engineering
The Australian National University
Canberra, Australia

Declaration

This thesis describes the results of research undertaken in the Department of Information Engineering, Research School of Information Sciences and Engineering, The Australian National University, Canberra. This research was supported by scholarships from the Australian Research Council under the Australian Postgraduate Awards (Industry) (APAI) and The Australian National University, Canberra.

The results and analyses presented in this thesis are my own original work, accomplished under the supervision of Doctor Roland Göcke, Doctor Hongdong Li and Doctor Nick Barnes, except where otherwise acknowledged. This thesis has not been submitted for any other degree.

Jason Saragih
Department of Information Engineering
Research School of Information Sciences and Engineering
The Australian National University
Canberra, Australia
1 October 2008

Acknowledgements

I would like to express my deep and sincere gratitude to my supervisor Dr. Roland Göcke for his unrivalled guidance and patience throughout the course of this study. His never-ending optimism and encouragement has been a guiding beacon in the rough-and-tumble of false starts and ideas that went pear-shaped. The lengths to which he helped was truly momentous and perhaps unprecedented. For this and more, I am truly grateful.

I would also like to thank my family for all their support and understanding during this period. To my parents, for providing space and shelter for me to pursue my studies. To my sisters, Vera and Melissa, for keeping my feet firmly grounded. To my brother, Jordan, for keeping me physically active as well as for the occasional comic relief. Also, to my cousins Vina and Meta, for the much needed nutrition.

My deepest gratitude also to Junko for her love, companionship and above all, patience, during this time. Her cheerful disposition and comforting demeanour proved irreplaceable though the numerous highs and lows.

Also, many thanks to my brothers and sisters in arms: Dave, Tal, Jill, Karl, Shawn and all those who joined in on getting down. Those were truly times to remember. Special thanks goes out to Dave and Tal, for their skewed personalities and hilarious perspectives, which made light of dilemmas and deflated misplaced immodesty.

I would also like to thank Dr. Hongdong Li, Dr. Nick Barnes and Prof. Richard Hartley for taking the time to humour my naive ideas. I found the numerous discussions we had to be enlightening and very instructive through the struggle to understand and go beyond.

Last but not least, I would like to thank Dr. David Austin for guiding me through the first years of my studies as well as giving me the opportunity to pursue this work to begin with.

Abstract

The recovery of a deformable visual object's structure from an image is a central problem in computer vision. It is often tackled through the utility of a Linear Deformable Model (LDM), which models variations of a visual object's shape and appearance linearly. This model has been shown to exhibit excellent modelling capacity whilst affording a compact representation of variability. However, it suffers from two major drawbacks. Firstly, there are significant difficulties regarding data collection, where a large number of correspondences is generally required in order to build the statistical models of shape and appearance that parameterise the LDM. The manual annotation of large databases can therefore be tedious and error prone. Secondly, approaches for structure recovery must address the conflicting goals of accuracy, reliability and efficiency.

In this thesis, contributions are made to address these two major areas of difficulty. In the first, the problem of automatic correspondence learning between pairs of images is tackled from a Bayesian perspective. The result is a general approach that allows domain knowledge to be integrated directly into the problem, where adaptations to similar problems are afforded through an explicit derivation of the involved components. In the second area of difficulty, the compromise between accuracy, reliability and efficiency in structure recovery is addressed through a generic method coined the iterative-discriminative approach. Leveraging on the predictive capacity of discriminative methods and the iterative framework of generative fitting, the approach is shown to exhibit excellent accuracy and reliability whilst also affording the most efficient procedure for LDM fitting known to date.

The problem of automatic correspondence learning is posed as a direct pairwise registration problem. Within its Bayesian formulation, it utilises the method of Hierarchical Priors in order to allow parameterisations of the involved densities to be optimised in conjunction with the correspondences. This is a significant step away from conventional approaches that utilise a fixed parameterisation, requiring a tedious cross validation procedure to determine the best parameterisation for a particular problem. Furthermore, the proposed approach introduces an objective criterion with which the quality of the found correspondences can be evaluated. Optimisation of the parameterisation and correspondences is achieved through the *marginalised maximum likelihood/maximum a posteriori* procedure that alternates between optimising the likelihood of the data with respect to the parameterisation (with marginalisation taken over the correspondences) and optimising the posterior of the correspondences for a fixed estimate of the parameterisation. The efficacy of the proposed approach is evaluated for the case of the human face on three types of databases: person specific, pose specific and generic person databases.

The iterative-discriminative approach for LDM fitting makes use of a novel fitting objective in its training procedure called *error bound minimisation*. This objective places emphasis on the gradual reduction of the spread of training samples about their respective optimum by

minimising the bound over the perturbations of the training data at each iteration. Since the objective only needs to be partially satisfied at each iteration, this approach allows simple regressors to be utilised, which exhibit better efficiency and generalisability in comparison to more complex ones. Four prototypes of the iterative-discriminative approach are proposed in order to tackle the problems of linear fitting, nonlinear fitting, robust fitting and background invariant fitting. The efficacy of the proposed prototypes is evaluated with regard to the problem of generic face fitting.

Finally, to facilitate further developments to the work presented in this thesis, implementations of the various proposed methods are provided along with this dissertation. The Deformable Model Library (`DeMoLib`), a C++ Application Programming Interface (API) for deformable model learning and fitting, provides a flexible software framework that builds on fixed parameterisations of the various flavours of LDMs, where extensions and developments in any aspect of their application can be easily augmented. This platform independent API is made publicly available for research purposes to encourage the timely dissemination of academic results.

Publications

During the course of this study, the following refereed conference papers were published.

- **Jason Saragih** and Roland Goecke, "A Nonlinear Discriminative Approach to AAM Fitting". In *Proceedings of the Eleventh IEEE International Conference on Computer Vision ICCV 2007*, Rio de Janeiro, Brazil, 14–20 October 2007. IEEE.
- **Jason Saragih** and Roland Goecke, "Monocular and Stereo Methods for AAM Learning from Video". In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR2007*, Minneapolis (MN), USA, 18–23 June 2007. IEEE.
- Shaun Press, **Jason Saragih** and Jason Chen, "Eye contact as a key component in Human Robot Interaction". In *Proceedings of the 2006 Australasian Conference on Robotics and Automation ACRA2006*, Auckland, New Zealand, 6–8 December 2006.
- **Jason Saragih** and Roland Goecke, "Learning Active Appearance Models from Image Sequences". In *Proceedings of the HCSNet Workshop on the Use of Vision in HCI VisHCI2006*, volume 56, pages 51-60, Canberra, Australia, 1–3 November 2006. ACS.
- **Jason Saragih** and Roland Goecke, "Iterative Error Bound Minimisation for AAM Alignment". In *Proceedings of the 18th International Conference on Pattern Recognition ICPR2006*, volume 2, pages 1192-1195, Hong Kong, China, 20–24 August 2006. IEEE.

Abbreviations

AAM	Active Appearance Model
ASM	Active Shape Model
ATLID	Asymptotically Trained Linear Iterative-Discriminative
BFGS	Broyden-Fletcher-Goldfarb-Shanno
BIID	Background Invariant Iterative-Discriminative
COTLID	Constrained Optimisation Trained Linear Iterative-Discriminative
C-RIC	Commensurate Robust Inverse Compositional
EM	Expectation Maximisation
FJ	Fixed Jacobian
GMM	Gaussian Mixture Model
HFBID	Haar-like Feature Based Iterative-Discriminative
ICA	Independent Component Analysis
ID	Iterative-Discriminative
KPCA	Kernel Principal Component Analysis
LDM	Linear Deformable Model
MAP	Maximum A Posteriori
MDL	Minimum Description Length
ML	Maximum Likelihood
MML	Marginalised Maximum Likelihood
MRI	Magnetic Resonance Imaging
NC-RIC	Non-Commensurate Robust Inverse Compositional
NIC	Normalised Inverse Compositional
PCA	Principal Component Analysis
PDF	Probability Density Function
POIC	Project-Out Inverse Compositional
PPCA	Probabilistic Principal Component Analysis
RAM	Random Access Memory
RGB	Red Green and Blue
RIC	Robust Inverse Compositional
RID	Robust Iterative-Discriminative
RMS	Root Mean Squared
SAT	Summed Area Table
SIC	Simultaneous Inverse Compositional
SVD	Singular Value Decomposition
SVR	Support Vector Regression
3DMM	3D Morphable Model

Contents

List of Figures

List of Tables

Introduction

The beginning is the end is the beginning.

Smashing Pumpkins

1.1 Motivation

Our world is not a rigid place. Many objects that we encounter in our daily lives exhibit inherent deformabilities. Understanding the deformations of these *deformable objects* has, therefore, proven vital in the advancement of many technological ventures.

Computer vision, a field that studies methods to understand images through the automatic recovery of their structure and its interpretation in the context of a problem, must therefore account for these deformations. In fact, due to the limited observatory power of images, even rigid 3D objects can appear to exhibit deformations in an image due to their projection onto the image as a *visual object*. Here, interpretation denotes the extraction of high level information from image structure, which defines the image's partitioning, functional properties and their relations to each other. For example, in the context of facial interpretation, this may involve the extraction of high level information such as: Is there a face in the image? Is it male or female? What is his/her emotional state? Who is it? In this case and many more, perhaps the most influential issue, which affects the possible deployment of computer vision applications on real world problems, is the recovery of the image's underlying structure, which can be thought of as a preprocessing step to image interpretation (see Figure 1.1). The deformations inherent in many *interesting* objects adds a degree of difficulty to structure recovery from images.

In the past, many attempts have been made that utilise only a coarse structure recovery process for image interpretation. Such methods, which generally utilise powerful and well developed machine learning algorithms such as Neural Networks and Support Vector Machines, embed a large proportion of the variations exhibited by these deformable visual objects into the interpretation process. Examples of these for face recognition can be found in [57; 69]. Although some impressive results have been reported using this approach, implementation difficulties inherent in these methods have restricted their usage for large scale deployment. One of the major sources of difficulty in this *holistic* interpretation approach is that deformations introduce nonlinearities into the visual object's appearance. For example, when structure recovery only involves the detection of an object's location and scale, the functional variation in pixel values within a rectangle containing a projected 3D object as it rotates follows a nonlinear appearance manifold in pixel space [95]. In order to gain sufficient accuracy for real world applications, there needs to be a large corpus of training data containing images of the object at an extensive range of poses. Although sufficiently large collections of training data are now available for many interesting problems, the extra nonlinearities caused by the inherent

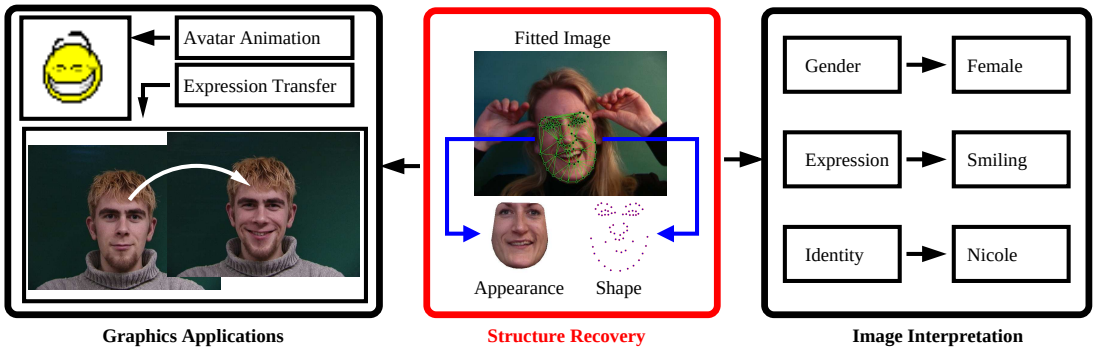


Figure 1.1: Structure recovery as a preprocessing step to image interpretation and graphics applications. Face images taken from the IMM Face Database [89].

variabilities of a visual object mean that interpretation usually involves a highly sophisticated nonlinear learner that can be computationally expensive to evaluate online. For real world applications that require a number of different interpretations of the same image, for example simultaneous visual speech and expression recognition, this approach can quickly become infeasible. Furthermore, the complexity of the predictive functions in the interpretation process often gives rise to generalisability problems.

In recent years, *deformable models* have enjoyed much attention in the computer vision community as a way to handle deformabilities of visual objects. This group of approaches utilises a more sophisticated structure recovery mechanism, where deformations are explicitly accounted for through model parameterisation. Deformation induced nonlinearities in the structure can then be accounted for by the interpretation process through structure normalisation. Throughout the years, some ingenious parameterisations and their utility have been proposed, such that the applicability of deformable models is now widespread in human-computer interaction [59; 138; 140], medical image analysis [98; 132; 148] and industrial vision [28; 39; 87].

The computer graphics community has also benefited from the development of deformable models. In this field, the recovered deformable structure is not used to normalise some data to be interpreted, but rather it is used explicitly for image synthesis. Examples of this include facial expression transfer [101; 136], avatar animation [77], visual speech synthesis [120] and face de-identification [56] (note that some of these applications involve a crossover with computer vision). In many applications in this field, the use of deformable models has allowed the automation of many tasks (see [19], for example), which previously required treatment by a human expert, significantly reducing workload as well as increasing efficiency.

Of particular interest in this thesis is a subclass of deformable models that will be referred to throughout this dissertation as the *linear deformable model* (LDM). Instances of this subclass have the distinction that they represent deformabilities, both in shape and appearance, as a linear object class (see Figure 1.2). Examples of LDMs include Active Shape Models (ASM) [31], Active Appearance Models (AAM) [30] and 3D Morphable Models (3DMM) [18]. LDMs recover structure from an image using the so called analysis-by-synthesis approach, whereby the LDM parameters are refined with the objective of attaining the best fit

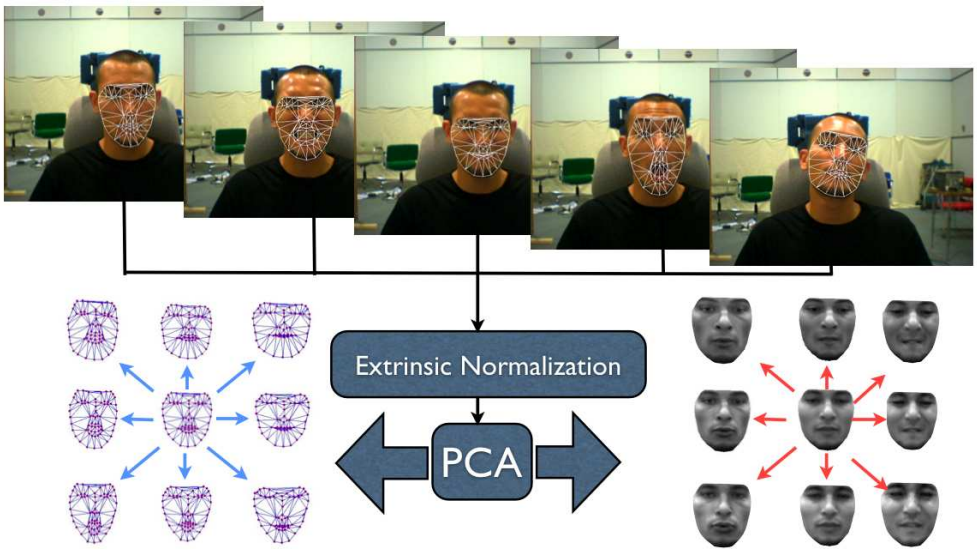


Figure 1.2: An illustration of the LDM learning process along with its various components. From a set of annotated training images, separate linear models of shape and appearance are learnt through principle component analysis (PCA).

between the synthesised model's appearance and that of the image (see Figure 1.3 for an illustration). Although any deformable visual object can be modelled by an LDM, it is particularly well suited to modelling visual objects whose variations live in a much smaller subspace than their representation. Examples of objects that exhibit this kind of variability include numerous anatomical structures, such as the human face [30]. One of the main strengths of LDMs is their compact representation of complex deformations by modelling the major directions of variations within the constrained subspace of variability.

1.2 Objectives

Despite enjoying an intense level of research over the last 15 years, LDMs still suffer from a number of limiting factors. Most notable amongst these are the difficulties in data collection for their training, and the trade-off between speed and accuracy in structure recovery (model fitting) as well as their robustness to occlusional effects and unmodelled variabilities. The first limiting factor arises from the way in which LDMs are parameterised, where deformabilities of the visual object of interest are represented using a statistical model of variations. These statistical models require a large number of correspondences to be available across a training dataset. Manual annotations of large datasets are both tedious and error-prone as well as lacking in repeatability. The second limiting factor is testament to the difficulty of deformable model fitting. Although LDMs are generally designed with an efficient parameterisation in mind, the number of parameters to be optimised in structure recovery can still be prohibitive for many applications. Most methods, therefore, make some assumptions in the model fitting procedure in order to improve computational efficiency. This, however, leads to reduced fitting accuracy as well as generalisability. Furthermore, these efficiency driven assumptions do not

Figure 1.3: An illustration of analysis-by-synthesis on an image taken from the IMM Face Database [89]. **Left to right**: Input image, initial model estimate, recovered structure after 5 iterations, recovered structure at convergence. Note that the RGB channels of the model are reversed (i.e. BGR) to highlight the model from the image.

generally account for occlusions or unmodelled variabilities, which are commonly encountered in real world applications, dramatically limiting the scope of these methods.

The primary goal of this thesis is to at least partially address the two major drawbacks of the LDM as described above. To address the problem of data collection, the utility of direct *pairwise* methods for automatic correspondence learning is rigorously investigated. Formulated within a Bayesian framework, a number of assumptions regarding the generative properties of deformable model matching, a component of the generative correspondence learning problem, as well as the distribution of their deformations, are investigated in a principled manner. The Bayesian framework adopted here also allows all the free parameters within the problem to be tuned automatically. This is a problem that has been largely ignored in most existing works. It will be shown that the regularised data fitting problem, which is the formulation often used in existing works, can be derived directly from a Bayesian formulation, and that it constitutes the case where the parameterisations of the densities involved in the Bayesian formulation are known and fixed. Through extensive empirical evaluations on the human face, the direct pairwise method for automatic correspondence learning is shown to be capable of modelling typical variations such as pose, lighting, expression and identity. However, it is also discovered that the method is highly sensitive to initialisation, where optimisation often terminates in a local minimum. Nonetheless, the Bayesian framework presented here serves as a flexible method from which further studies can benefit. An example of the adaptation of the proposed procedure as a groupwise method, where the linear models of the LDM's shape and appearance are learnt along with the correspondences, is also presented in this dissertation.

To address the usual trade-off between speed and accuracy in deformable model fitting, a new fitting approach for LDMs is introduced. The approach is specifically designed for flexibility to accommodate the two opposing criteria of a fitting algorithm: the accuracy requirements of a problem and the computational capacity of the system that implements it. This is a major shift in paradigm from the general attitude of either building the most powerful model possible with the expectation of increases in computational power in the future [105], or applying some approximations in order to facilitate a reduced computational burden [30]. This coupling of desired accuracy and computational cost provides system engineers greater flexibility in designing and planning the construction of integrated systems that utilise this fitting procedure. The approach leverages on the efficiency and generalisation properties of

discriminative methods. Training on simulations of real fitting problems, it is shown that this approach exhibits excellent generalisability on unseen instances of the visual object as well as affording a flexibility in the level of desired accuracy, which can be tuned based on the needs of the image interpretation application that uses its recovered structure. This is achieved through the concept of iterative error bound minimisation, whereby at each iteration of the algorithm computational resources are focused primarily on tackling the worst case scenarios, minimising the errors on simulated samples that are furthest from their desired settings. By virtue of its iterative framework, the discriminative predictors (regressors) need only partially satisfy the problem's objective at each iteration, since the continuity of objective between iterations gives rise to further overall improvements in future iterations. As such, the approach affords the utilisation of simple functional forms for its predictors, which generally exhibit better generalisability than their more complex counterparts, as well as affording a rapid evaluation. The approach proposed here is also highly applicable, as instances can be created using a variety of model parameterisations, regressors and feature extraction procedures (that are used to drive the regressors). As such, a number of prototypes of this approach will be evaluated in this dissertation, highlighting its applicability. These prototypes include those that utilise linear and nonlinear regressors as well as one that is robust in the presence of occlusional effects. An extension of the linear prototype that can handle varying backgrounds is also presented, where it is shown that background invariance can be achieved without sacrificing performance.

A secondary goal of this dissertation is to provide a flexible software framework that builds on fixed parameterisations of the various flavours of LDMs, where extensions and developments in any aspect of their application can be easily augmented. For this, the Deformable Model Library (`DeMoLib`), a C++ Application Programming Interface (API) for deformable model learning and fitting, is provided along with this dissertation. A number of components commonly used by LDMs can be found here, such as linear shape and appearance model classes, warping and various other geometric functions (Procrustes alignment, for example), as well as full implementations of a number of prominent AAM and ASM fitting procedures. The API also provides a Graphical User Interface (GUI) for a number of common tasks, such as manual annotation, linear model viewing, and visualisation of model fitting and tracking procedures. Although a number of similar libraries now exist, most have their drawbacks. The AAM API [113], for example, implements only the original AAM fitting procedure and is platform dependent (i.e. it is a Windows only API). Another example is `am_tools`[1], for which the source code is not publicly available. In contrast, `DeMoLib` is a platform independent API whose source code is made publicly available for research purposes. Finally, it should be noted that all experiments presented in this dissertation were implemented using `DeMoLib`, allowing reproduction of all results using the publicly available database on which the experiments were conducted.

1.3 Overview

This dissertation is comprised of six chapters, the first of which is this introduction. The chapters are organised in such a way that the reader will benefit by reading the chapters in

[1] `http://www.isbe.man.ac.uk/~bim/software/am_tools_doc/index.html`

order as conventions and terminology set out in earlier chapters are adopted in the chapters that follow. This is especially the case for Chapters 4 and 5, where the latter is an empirical evaluation of the former. However, the problems tackled in Chapters 3 and 4, as well as their proposed solutions, are separate and distinct. As such, the reader can freely interchange the order of these chapters, but is strongly encouraged to first read Chapter 2.

A brief outline of each of the chapters that follow is given below:

Chapter 2 comprises a general overview of LDMs. This includes a detailed discussion of their common parameterisations and a brief outline of some less common representations. The models described in this chapter serve as a basis for the prototypes used in the experiments of LDM fitting, presented in Chapter 5. A review of existing methods for automatic correspondence learning for LDM building is also presented, where the strengths and weaknesses of some of the more prominent methods are discussed. Finally, a taxonomy of existing LDM fitting approaches is presented, grouping the methods based on their algorithmic realisations.

Chapter 3 presents a rigorous investigation into the utility of direct pairwise approaches for automatic correspondence learning. The formulation of the problem within a Bayesian framework is derived along with a discussion of possible parameterisations for the involved densities. The applicability of the approach is empirically evaluated through experiments on a face database, testing its performance for person specific, pose specific and generic person models. Analysis of the results is presented along with suggestions for further improvements.

Chapter 4 presents the *iterative-discriminative* approach for LDM fitting, a novel approach that leverages on the predictive capacity of discriminative methods and the iterative framework of generative fitting, coupled through the objective of error bound minimisation. Details regarding its derivation as well as the motivating factors involved are discussed with reference to existing fitting approaches. Several prototype methods are presented that utilise linear and nonlinear regressors as well as extensions that can handle occlusions and varying backgrounds.

Chapter 5 comprises an investigation into the efficacy of the iterative-discriminative approach through experiments on the various proposed prototypes. Empirical evaluations are performed on the difficult problem of generic face fitting with comparisons made against a number of existing methods for LDM fitting. Analyses of the results are presented along with ideas for further performance gains.

Chapter 6 concludes this dissertation with an overview of contributions and mention of directions for future work.

1.4 Mathematical Nomenclature

In order to facilitate a better understanding of the mathematical formulae in this dissertation, conventions on notations used throughout this thesis are presented below.

Scalars are written in italics, either in lower or upper-case, for example: a and B.

Vectors are written in lower-case non-italic boldface, with components separated by spaces, for example:

$$\mathbf{v} = \begin{bmatrix} a & b & c \end{bmatrix}^T = [a\ ;\ b\ ;\ c] = \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \tag{1.1}$$

is a column vector, where ^{T} denotes the vector or matrix transpose. The type of brackets is chosen in the context of an equation to clarify the exposition. Elements of a vector are represented by the lower-case italic vector name with a parenthesised index as its subscript. For example: $v_{(i)}$ is the i^{th} element of vector $\mathbf{v}$. The size of a vector is represented by a parenthesised number as its superscript, for example: $\mathbf{v}^{(n)}$ is an n-length vector. Sub-vectors are represented by the range in the indices, for example: $\mathbf{v}_{(2:5)}$ denotes the 4-length vector comprising of elements 2 to 5 of $\mathbf{v}$ inclusive. If no starting or ending index is specified, then the sub-vector consists of elements to the beginning or end of the vector, respectively, for example: $\mathbf{v}_{(7:)}$ comprises all elements of $\mathbf{v}$ from the 7^{th} element onwards.

Matrices are written in upper-case non-italic boldface, for example:

$$\mathbf{M} = [\ a\ \ b\ ;\ c\ \ d\ ] = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}. \tag{1.2}$$

The type of brackets is chosen in the context of the equation to clarify the exposition. Elements of a matrix are represented by the italic, upper-case matrix name with a parenthesised index as their subscript, for example: $M_{(i,j)}$ is the element in the i^{th} row and j^{th} column. The size of the matrix is represented by its parenthesised superscript, for example: $\mathbf{M}^{(n\times m)}$ is a matrix with n rows and m columns. Sub-matrices are represented by the range in the indices, for example: $\mathbf{M}_{(2:5,1:3)}$ denotes the (4×3) matrix comprising the second to fifth rows of $\mathbf{M}$ and its first to third column inclusive. If no endpoints are set, then the sub-matrix consists of elements to the end of that column or row, for example: $\mathbf{M}_{(1,:)}$ denotes the first row of $\mathbf{M}$.

Vector diagonalisation is represented by the diag$\{.\}$ operator, where each element of the vector is placed in the diagonal entries of the matrix, for example:

$$\text{diag}\{\mathbf{v}\} = \text{diag}\{[a\ ;\ b\ ;\ c]\} = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix}. \tag{1.3}$$

Vector of constants are typeset as the boldface of the number, for example: $\mathbf{1} = [1\ ;\ \ldots\ ;\ 1]$ or $\mathbf{0} = [0\ ;\ \ldots\ ;\ 0]$.

Inner product of two vectors is represented by the $\langle ., . \rangle$ operator, for example:

$$\langle \mathbf{v}, \mathbf{w} \rangle = \mathbf{v}^T \mathbf{w}. \tag{1.4}$$

Matrix vectorisation is represented by the vec$\{.\}$ operator, which takes each column of a matrix and concatenates them into a vector, for example:

$$\text{vec}\{\mathbf{M}\} = \text{vec}\{[\ a\ \ b\ ;\ c\ \ d\ ]\} = [a\ ;\ c\ ;\ b\ ; d]. \tag{1.5}$$

Matrix determinant is represented by the det$\{.\}$ operation.

Kronecker product is represented by the $\otimes$ symbol, for example:

$$\mathbf{M} \otimes \mathbf{N} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \otimes \mathbf{N} = \begin{bmatrix} a\mathbf{N} & b\mathbf{N} \\ c\mathbf{N} & d\mathbf{N} \end{bmatrix}. \tag{1.6}$$

Identity matrices are typeset as:

$$\mathbf{I} = \begin{bmatrix} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1 \end{bmatrix}. \tag{1.7}$$

Sets are typeset using curly brackets: $\{a, b, c\}$ or $\{\mathbf{x}_i\}_i^N$.

Spatial set within a triangle is denoted by tri$\{\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k\}$, where the triangle vertices are $\mathbf{x}_i, \mathbf{x}_j$ and $\mathbf{x}_k$.

Spatial set within a convex hull is denoted by hull$\{\mathbf{s}\}$, where $\mathbf{s} = [\ x_1\ ;\ y_1\ ;\ \dots\ ;\ x_n\ ;\ y_n\ ]$ is a vector containing the 2D points defining the convex hull.

Functions are typeset in the upper-case Ralph Smith's Formal Script (RSFS) font, for example: $\mathscr{F}(\mathbf{x}; \mathbf{p})$. Here, $\mathbf{p}$ are the variables of $\mathscr{F}$ and $\mathbf{x}$ are the dependents.

Function composition is denoted by the the $\circ$ symbol, for example:

$$\mathscr{F}(\mathscr{G}(\mathbf{x}; \mathbf{v})) = \mathscr{F} \circ \mathscr{G}(\mathbf{x}; \mathbf{v}). \tag{1.8}$$

When composing functions with multiple variables, the variable resulting from the evaluation of the composed function is set as the diamond symbol (i.e. a place holder):

$$\mathscr{F}(\mathscr{G}(\mathbf{x}; \mathbf{v}); \mathbf{p}) = \mathscr{F}(\diamond; \mathbf{p}) \circ \mathscr{G}(\mathbf{x}; \mathbf{v}). \tag{1.9}$$

Expectation of a function is denoted:

$$E_{p(\mathbf{x})}\left[\ \mathscr{F}_{\mathbf{x}}\ \right], \tag{1.10}$$

where the expectation is taken with respect to the probability density function $p(\mathbf{x})$.

Chapter 2

Linear Deformable Models

... and I've seen it before
... and I'll see it again
... yes I've seen it before
... just little bits of history repeating.

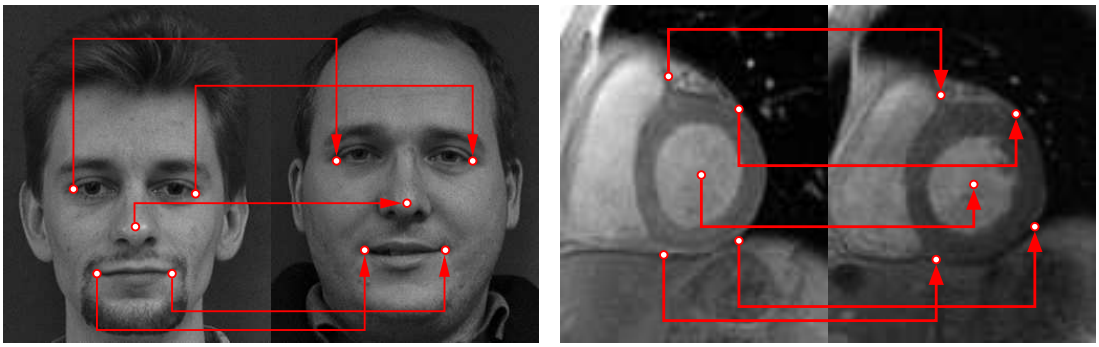
Propellerheads

The Linear Deformable Model (LDM) is perhaps one of the most common mathematical tool used to represent deformable visual objects. The computer vision community started utilising this model for use in analysis-by-synthesis type problems in the early 1990s. Since then, significant advances have been made in improving their representative power and the computational efficiency of their use, as well as opening up new domains of application.

In this chapter, a detailed review of LDMs is presented. Aspects pertaining to the various parameterisations of its different flavours are discussed in Section 2.1, concentrating on the representation of both its shape and appearance as a linear object class. Existing approaches for automatic correspondence learning and model building, the first area to which this dissertation contributes, are reviewed in Section 2.2. The various existing approaches to LDM fitting, the second topic on which this dissertation contributes, are discussed in Section 2.3, where approaches are grouped according to their algorithmic realisations. This chapter concludes in Section 2.4 with an overview and a brief discussion of related topics.

2.1 Parameterisation

There currently exist a number of different flavours of LDMs in the literature, each of which is specialised to a particular type of visual object. For example, the Active Shape Model (ASM) was designed to model visual objects with strong boundary features, such as the outline of a human hand and bones in medical images, the Active Appearance Model (AAM) was designed to handle objects that exhibit a large amount of appearance variation within its class and the 3D Morphable Model (3DMM) extends the AAM's representative power to the 3D surface domain, explicating the true dimensionality of the object being modelled as well as affording a higher fidelity in detail. Despite their apparent differences, under the guise of slightly different names and acronyms, their underlying mathematical framework is very similar. However, they differ in their fitting procedure. One of the main common factors amongst the various LDM flavours, is their intrinsic representation of shape and texture as a linear object class.



(a) (b)

Figure 2.1: Homologous point set: Corresponding points across different images relating the same physically meaningful location. **(a):** Example of homologous points for the human face taken from the IMM Face database [89]. **(b):** Example of homologous points for the left ventricle with images taken from [112].

2.1.1 Parameterising Shape

The shape of an LDM, whether describing a 3D object or a 2D visual object, is generally represented by a set of points $\{\{\mathbf{x}_i\}_i^n | \mathbf{x}_i \in \Re^{D_s}\}$, where D_s is the dimensionality of the model points (2D or 3D). This is in contrast to the representation of more general deformable models, which represent shapes by functionals such as curves, circles, or Fourier descriptors specific to the particular object being modelled [111; 143]. The points $\mathbf{x}_i$ in an LDM, commonly coined *landmarks*, are often chosen to correspond to physically meaningful locations on the visual object, which are consistently located in any instance within the visual object's class. An example is the outer corner of the eye for the visual object class of human faces (see Figure 2.1). Despite the various landmark configurations, defined by the set of points $\{\mathbf{x}_i\}_i^n$ for each face, the location of a landmark $\mathbf{x}_i$ always corresponds to the same physical point in all faces. Although the landmarks, and hence the physically meaningful points, can be chosen arbitrarily, in practice, points corresponding to salient visual features, such as corners and edges, are most often used as they allow more reliable manual annotations.

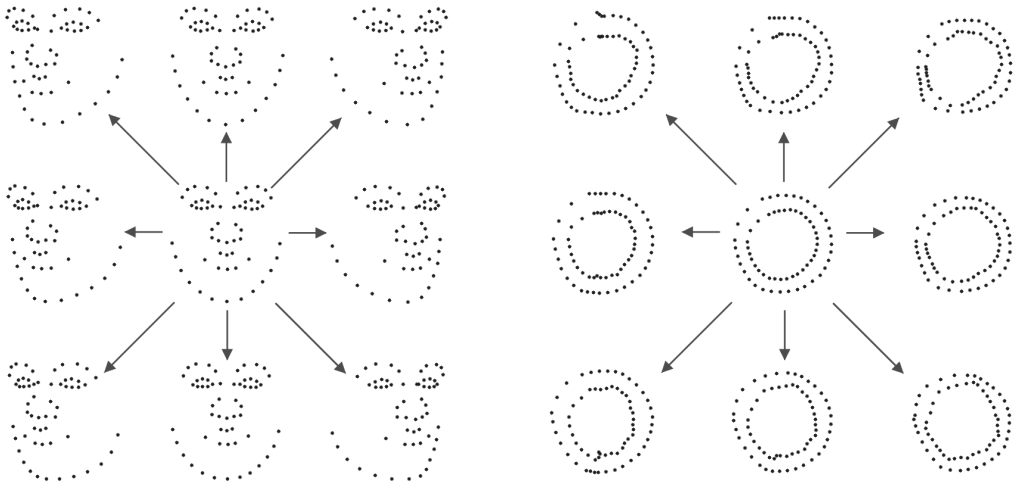
For mathematical treatment, the shape of an LDM is usually represented as a $(D_s n)$-length vector, consisting of an ordered concatenation of the individual landmarks:

$$\mathbf{s} = [\, \mathbf{x}_1 \; ; \; \ldots \; ; \; \mathbf{x}_n \,], \tag{2.1}$$

where n is the number of landmarks defining the visual object's shape. Rather than directly parameterising the visual object's shape through landmark locations, LDMs afford a much more compact representation that is decomposed into intrinsic and extrinsic accounts of shape variability.

Intrinsic Shape Variation

The intrinsic or local shape variation of LDMs generally accounts for shape deformabilities that are independent of the imaging conditions. These deformations are accounted for here by



(a) (b)

Figure 2.2: (a): Example of the first two modes of intrinsic shape variation of a human face built using the IMM Face database [89]. **(b):** Example of the first two modes of intrinsic shape variation of the left ventricle built using the database described in [112]. Each mode of variation is varied between ± 3 standard deviations of the mean shape, keeping the other intrinsic parameters at zero.

a linear combination of modes of variation:

$$\mathscr{S}_l(\mathbf{p}_s)\colon \Re^{M_s} \rightarrow \Re^{D_s n} = \bar{\mathbf{s}} + \mathbf{\Phi}_s \mathbf{p}_s \tag{2.2}$$

where $\mathscr{S}_l$ is the intrinsic shape generating function, $\bar{\mathbf{s}}^{(D_s n)}$ is the mean shape, $\mathbf{\Phi}_s^{(D_s n \times M_s)}$ is a matrix of concatenated modes of intrinsic shape variation and $\mathbf{p}_s^{(M_s)}$ are the intrinsic shape parameters, which define coordinates within the subspace spanned by $\mathbf{\Phi}_s$. An example of intrinsic shape variation is illustrated in Figure 2.2. This representation is appropriate for deformable objects where the distribution of the shapes can be adequately approximated by a low-rank or degenerate Gaussian. Examples of objects that have previously been successfully represented in this way include the human face [43; 18] and numerous other anatomical structures [32]. Representing objects using a linear model, where the distribution of the elements of $\mathbf{p}_s$ do not follow that of a Gaussian or uniform distribution, can result in shape instantiations that are not physically realisable. Examples of this include objects with rotating components or those exhibiting significant 3D view changes [103].

For many visual objects, the number of modes of variation M_s is much smaller than the size of the shape vector $D_s n$, resulting in a compact representation for modelling intrinsic shape variability. These modes of variation are commonly found through the application of Principle Component Analysis (PCA) on a set of extrinsically aligned shapes $\{\tilde{\mathbf{s}}_i\}_i^N$ (see Section 2.1.1), retaining only the subset of modes that account for the majority of variation within the set.

Applying Singular Value Decomposition (SVD) to the covariance matrix:

$$\mathbf{C} = \frac{1}{N-1}\sum_{i=1}^{N}(\tilde{\mathbf{s}}_i - \bar{\mathbf{s}})(\tilde{\mathbf{s}}_i - \bar{\mathbf{s}})^T = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^T \quad \text{where} \quad \bar{\mathbf{s}} = \frac{1}{N}\sum_{i=1}^{N}\tilde{\mathbf{s}}_i, \tag{2.3}$$

the modes of variation are generally chosen as the M_s eigenvectors corresponding to the M_s-largest eigenvalues:

$$\mathbf{\Phi}_s = \mathbf{U}_{(:,1:M_s)} \quad \text{where} \quad \left\{\forall\, i < j \colon \mathbf{\Sigma}_{(i,i)} \geq \mathbf{\Sigma}_{(j,j)}\right\}. \tag{2.4}$$

The choice of M_s is something of a 'black art' that often depends on other criteria imposed on the model. Listed in the following are a few common approaches to its selection:

- If the variance of noise σ^2 in the estimates of $\tilde{\mathbf{s}}$ is known, then M_s is set to the maximum number such that $\Sigma_{(M_s,M_s)} > \sigma^2$.

- Find the *knee* in the eigenspectrum of $\mathbf{C}$. However, in many problems, a clear decrease in the eigenspectrum between the last mode of variation and noise is not easily distinguishable. Typically, the eigenspectrum of real datasets tend to taper off smoothly (see Figure 2.3). This is particularly the case for visual objects for which a truncated linear model is an approximation.

- Set a required reconstruction accuracy and increase M_s until the required accuracy over every shape in the training set is achieved. This approach requires significant domain knowledge, both of the visual object and the fitting regime for which it will be used. Alternatively, a cross-validation procedure can be utilised, whereby the dataset is partitioned into training and test sets. M_s can then be incrementally increased until the model overlearns the data, which can be determined by an increase in the reconstruction error on the test set. However, this procedure can be computationally expensive, especially for the appearance model that requires similar treatment (see Section 2.1.2).

- Utilising parallel analysis, the data's eigenspectrum is compared to the eigenspectrum of a randomised version of the data [114]. Although this approach requires no domain knowledge, it has a tendency to underfit the data.

- Assume a certain proportion of the total variation in the training set is due to noise:

$$\frac{\sum_{i=1}^{M_s}\mathbf{\Sigma}_{(i,i)}}{\sum_{i=1}^{D_s n}\mathbf{\Sigma}_{(i,i)}} \geq d\% \tag{2.5}$$

  Here, d is commonly chosen to be a fairly large proportion, such as 95% or 98%.

In practice, by far the most popular out of these is the last method, which is sufficient for many cases.

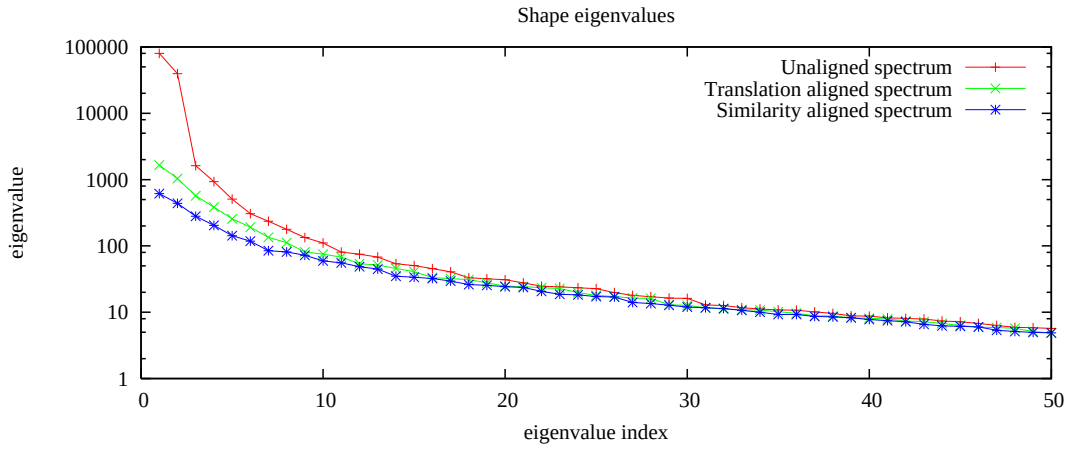


Figure 2.3: Logarithmic plot of the eigenspectrum of a linear shape model, built from non-aligned and aligned training shapes. The alternating Procrustes alignment method was used for alignment.

Extrinsic Shape Variation

To facilitate a compact model of intrinsic shape variations, the effects of extrinsic (global) shape variations must be accounted for separately. These extrinsic variations account for the different geometrical conditions under which the visual object is observed. It can be thought of as the projection of the intrinsic shape, defined in the *model frame*, onto the *image frame*. This projection consists of a composition of the intrinsic shape generating function $\mathscr{S}_l$ with the projection function:

$$\mathscr{S}(\mathbf{p}_s, \mathbf{g}_s) \colon \Re^{M_s+G_s} \rightarrow \Re^{D_s n} = \mathscr{S}_g(\diamond; \mathbf{g}_s) \circ \mathscr{S}_l(\mathbf{p}_s), \tag{2.6}$$

where $\mathscr{S}_g$ is the projection function, parameterised by $\mathbf{g}_s^{(G_s)}$.

For 2D LDMs, the projection function is generally chosen as the similarity transform:

$$\mathscr{S}_g(\mathbf{s}; \mathbf{g}_s) \colon \Re^{G_s} \times \Re^{2n} \rightarrow \Re^{2n} = \left(\mathbf{I}^{(n \times n)} \otimes \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \right) \mathbf{s} + \mathbf{1}^{(n)} \otimes \begin{bmatrix} t_x \\ t_y \end{bmatrix}, \tag{2.7}$$

where $\mathbf{g}_s = [\ a\ ;\ b\ ;\ t_x\ ;\ t_y\ ]$. Here, a and b define the parameterisation of a scaled rotation matrix, with:

$$a = s\ \cos(\theta) \quad \text{and} \quad b = s\ \sin(\theta), \tag{2.8}$$

where s and θ denote the scale factor and rotation angle, respectively. It should be noted here, that in some works, such as [45; 139], the parameterisation of the shape generation function is simplified by extending the linear intrinsic variations to account for the extrinsic variations. This is achieved by concatenating $[\bar{x}_1; \bar{y}_1; \ldots; \bar{x}_N; \bar{y}_N]$, $[-\bar{y}_1; \bar{x}_1; \ldots; -\bar{y}_N; \bar{x}_N]$, $[1; 0; \ldots; 1; 0]$ and $[0; 1; \ldots; 0; 1]$ to the columns of $\mathbf{\Phi}_s$ in Equation (2.2). Approximating the similarity transform in this way does not apply the rotations and scalings to the linear modes of intrinsic variation, only to the mean shape. Although the unmodelled scaling in the intrinsic variations can be accounted for by directly scaling the parameters $\mathbf{p}_s$, since the rotations

are not modelled, the expressive power of this parameterisation is limited. Furthermore, the combination of a rotated mean with an unrotated basis can result in implausible shapes.

For 3D LDMs, the projection function takes the form of a 3D projection, or one of its various approximations. Shown below is the weak-perspective projection model commonly used in 3DMMs:

$$\mathscr{S}_g(\mathbf{s};\mathbf{g}_s)\colon \Re^{G_s}\times\Re^{3n}\rightarrow\Re^{2n}=\left(\mathbf{I}^{(n\times n)}\otimes s\mathbf{R}\right)\mathbf{s}+\mathbf{1}^{(n)}\otimes[\ t_x\ ;\ t_y\ ], \tag{2.9}$$

where $\mathbf{g}_s = [\ s\ ;\mathrm{vec}(\mathbf{R})\ ;\ t_x\ ,\ t_y\ ]$. Here, $\mathbf{R}^{(2\times 3)}$ contains the first two columns of a rotation matrix.

Extrinsic Alignment

As the training set $\{\mathbf{s}\}_i^N$ generally consists of annotations in the image frame, they must first be *aligned* before applying PCA to obtain a linear shape model, in order to minimise the effects of extrinsic shape variations from the training set. An appropriate objective to optimise is the *compactness* of the linear model built from the aligned shapes. Compactness is most effectively measured by the number of modes of intrinsic shape variation M_s. However, since the amount of noise in the annotations is generally unknown, it is difficult to apply this measure in practice.

One of the most common extrinsic alignment methods is an iterative approach utilising Procrustes alignment [52] to align each shape to the mean image, then recomputing the mean, repeating these alternating steps until some convergence criterion is met. However, since Procrustes alignment assumes an isotropic error on each point in alignment, this procedure may result in a biased estimate that does not achieve optimal compactness. Another solution is to iteratively learn the model, interleaving model building and fitting steps. However, fitting a linear model with extrinsic variations composed is a nonlinear process, increasing the likelihood of the procedure terminating in a local minimum. Recently, a linear closed form solution to the problem of automatic intrinsic and extrinsic model extraction was proposed in [142]. The method requires M_s to be set *a-priori* and uses the basis constraint to make the problem well posed. However, concerns regarding the robustness of this method in the presence of measurement noise was expressed in [21], requiring the correct M_s to be used to obtain accurate results. This problem stems from the maximum-likelihood framework from which the linear solution was derived, which places no prior on the intrinsic shape parameters.

Nonetheless, the simple alternating procedure described above has been used widely for shape alignment and gives sufficiently accurate alignment for obtaining a reasonably compact shape model in many scenarios. It should be noted here, that even with poor extrinsic alignment, the resulting model may still be useful, despite some of the extrinsic variation being modelled in the intrinsic linear model. Figure 2.3 illustrates the utility of extrinsic shape alignment for compact linear shape model building. Here, alignment is achieved using the tangent space alignment method [43], where each shape is transformed to the tangent space of the mean. Note that the similarity aligned model exhibits a more compact spectrum compared to the translation aligned model, which in turn is more compact than the model built from unaligned shapes.

2.1.2 Parameterising Appearance

The appearance model of an LDM represents how the visual object of interest appears in an image. Its utility here is twofold. First and foremost, it is generally used to measure the fit between an image and the model at its current parameter settings (see Section 2.3.2). The second utility is a graphics one, in which instances of the object can be synthesised for animation-type applications (see [121], for example). The appearance model of an LDM can incorporate a large amount of information about the visual object, such as multi-plane representations (i.e. RGB images), processed image pixels (i.e. Gabor wavelets) and voxel values for 3D LDMs. These representations generally depend on the type of visual object as well as the intended application of the model.

Regardless of the types of features used, an instance of the LDM's appearance is generally represented as a vectorised image:

$$\mathbf{a} = [\, \mathbf{v}_1 \; ; \; \ldots \; ; \; \mathbf{v}_P \,], \tag{2.10}$$

where $\mathbf{v}_i^{(D_a)}$ denotes the appearance of the i^{th} pixel out of P, in a model with D_a imaging planes. To maintain a fixed number of pixels over all model instances, for ease of mathematical treatment, the appearance is generally defined for locations within a prespecified region $\mathbf{\Omega}$ in the so called "canonical frame". For the AAM and 3DMM, $\mathbf{\Omega}$ is generally defined as the set of all pixels within the convex hull of a predefined shape, where by convention the mean shape $\bar{\mathbf{s}}$ is often used. Other methods, such as the ASM or the Active Feature Model [67], utilise a local appearance representation around each of the shape's landmarks in this frame[1]. To evaluate the fitting quality of a particular configuration of the LDM's parameters, the image is *cropped* onto the canonical frame through the utilisation of a warping function:

$$\mathscr{W}(\mathbf{x}; \mathbf{s}) \colon \Re^2 \times \Re^{D_s n} \rightarrow \Re^2, \tag{2.11}$$

that denotes the location of a pixel in the canonical frame, projected into the image frame, expressed through the current shape $\mathbf{s}$ in the image frame. For appearance synthesis, the inverse of $\mathscr{W}$ is utilised. Figure 2.4 illustrates the process of appearance cropping and synthesis. The type of warping function to be used here will generally depend on the type of visual object being modelled. However, most instances of LDM's utilise a fixed type of function, regardless of the object being modelled. For example, the AAM utilises the piecewise affine warp, the 3DMM utilises a direct interpolation function (due to its dense shape representation), and the ASM utilises a profile extraction function. As with the shape model described in Section 2.1.1, the appearance model is also composed of intrinsic and extrinsic variations. In the following, each of these sources of variation are discussed in turn.

Intrinsic Appearance Variation

The intrinsic or local appearance model of an LDM accounts for changes in the visual object's appearance, which are independent of imaging conditions. As with intrinsic shape variations,

[1] Note that this kind of appearance representation is used primarily for fitting rather than synthesis.

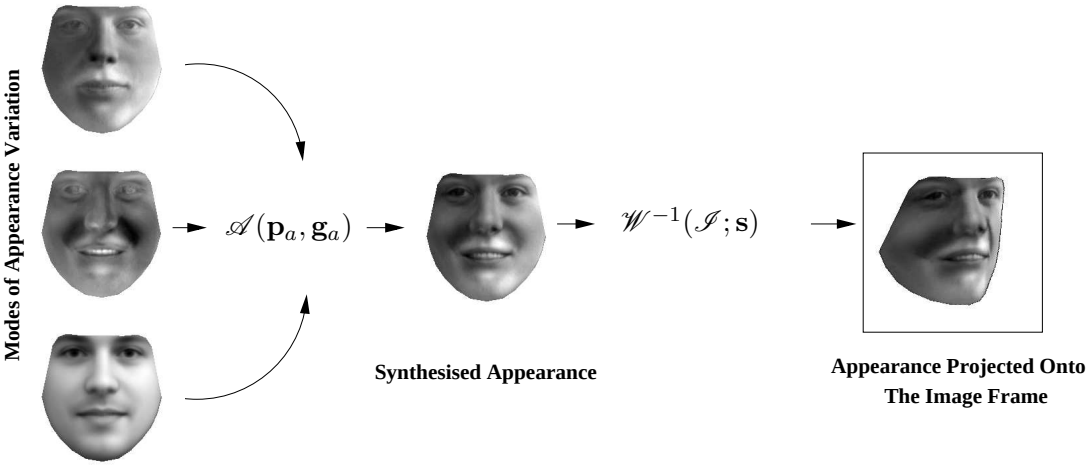


Figure 2.4: Illustration of appearance synthesis in an LDM.

the appearance variations are also represented by a linear combination of modes of variation:

$$\mathscr{A}_l(\mathbf{p}_a)\colon \Re^{M_a} \rightarrow \Re^{D_a P} = \bar{\mathbf{a}} + \mathbf{\Phi}_a \mathbf{p}_a \tag{2.12}$$

where $\mathscr{A}_l$ is the intrinsic appearance generating function, $\bar{\mathbf{a}}^{(D_a P)}$ is the mean appearance, $\mathbf{\Phi}_a^{(D_a P \times M_a)}$ is a matrix of concatenated modes of intrinsic appearance variation and $\mathbf{p}_a^{(M_a)}$ are the intrinsic appearance parameters. An example of intrinsic appearance variation is illustrated in Figure 2.5.

The procedure for obtaining the intrinsic appearance model is the same as that for shape, described in Section 2.1.1. The main difference here concerns the dimensionality of the appearance vector $\mathbf{a}$. Since the number of pixels within $\mathbf{\Omega}$ is generally much larger than the number of available training images (a notable exception being the ASM's representation), directly performing SVD on the covariance matrix will, in general, be extremely costly. As such, an alternate approach is often utilised. Let the covariance matrix be written as:

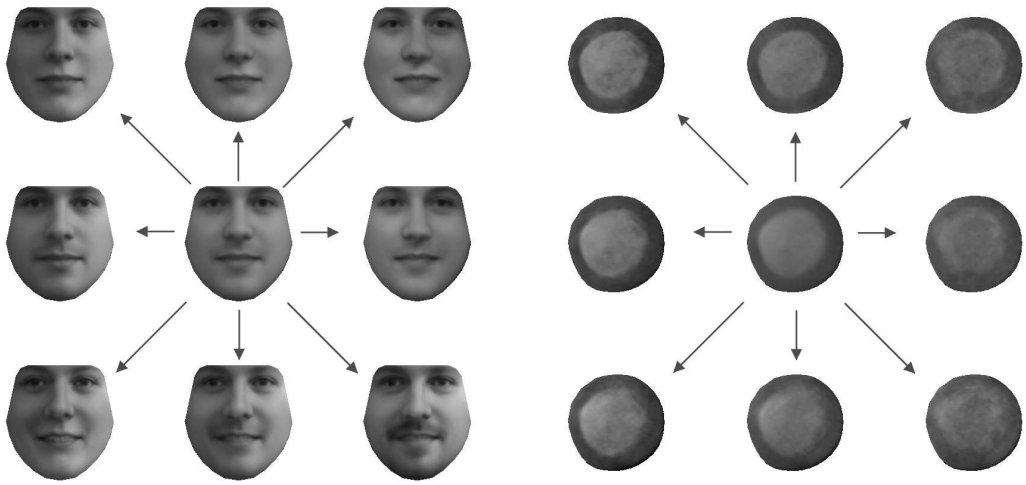
$$\mathbf{C} = \frac{1}{N}\mathbf{A}\mathbf{A}^T \quad \text{where} \quad \mathbf{A} = \begin{bmatrix} \tilde{\mathbf{a}} - \bar{\mathbf{a}} & \dots & \tilde{\mathbf{a}} - \bar{\mathbf{a}} \end{bmatrix}. \tag{2.13}$$

Here, $\tilde{\mathbf{a}}$ is the extrinsically normalised cropped image. Since $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$ share the same non-zero eigenvalues [86], and the eigenvectors of $\mathbf{A}\mathbf{A}^T$ corresponding to these eigenvalues are related to the eigenvectors of $\mathbf{A}^T\mathbf{A}$ through:

$$\mathbf{\Phi}_a = \mathbf{A}\hat{\mathbf{\Phi}}_a \quad \text{where} \quad \mathbf{A}^T\mathbf{A} = \hat{\mathbf{\Phi}}_a \mathbf{\Lambda} \hat{\mathbf{\Phi}}_a^T = \hat{\mathbf{\Phi}}_a \,\text{diag}([\,\lambda_1\,;\,\dots\,;\,\lambda_N\,])\,\hat{\mathbf{\Phi}}_a^T, \tag{2.14}$$

then the non-zero eigenvalues of the covariance matrix and their corresponding eigenvectors can be computed by performing SVD on the smaller $(N \times N)$ matrix $\mathbf{A}^T\mathbf{A}$. Note that when using this approach, the columns of $\mathbf{\Phi}_a$ may require re-normalising since they will not, in general, be of unit length.

In the more general case, when the number of images N is very large, performing SVD on



(a) (b)

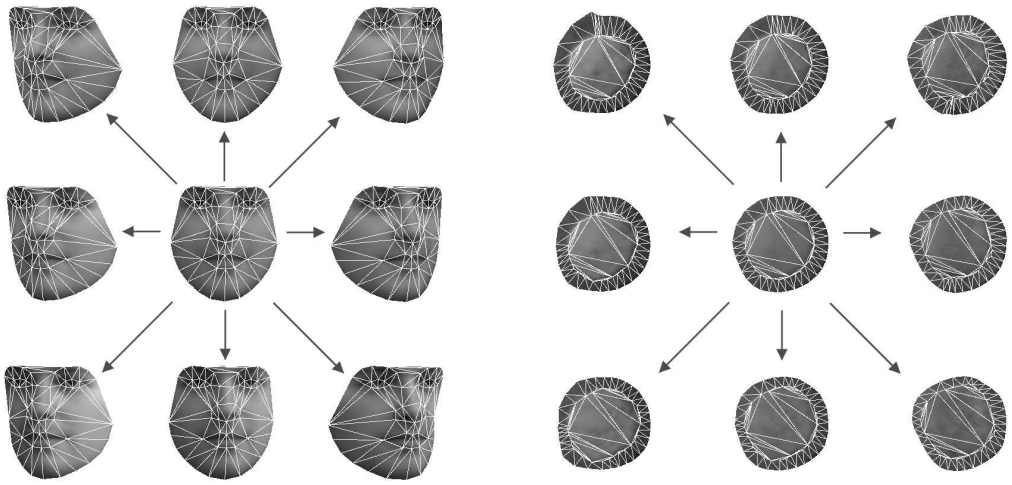
Figure 2.5: (a): Example of the first two modes of intrinsic appearance variation for a human face learnt from the IMM Face database [89]. **(b):** Example of the first two modes of intrinsic appearance variation for the left ventricle learnt from the database described in [112]. Each mode of variation is varied between ± 3 standard deviations of the mean shape, keeping the other intrinsic parameters at zero.

$\mathbf{A}^T\mathbf{A}$ may still be intractable. In such cases, methods for incremental SVD must be employed. The method proposed in [20], which decomposes the matrix $\mathbf{A} \leftarrow \mathbf{U}\mathbf{S}^{\frac{1}{2}}\mathbf{V}$ by incrementally adding one column of $\mathbf{A}$ to the equation system. The resulting eigenvalues of $\mathbf{A}$ are then the positive square roots of the nonzero eigenvalues of $\mathbf{A}\mathbf{A}^T$, and the left-hand singular vectors $\mathbf{U}$ of $\mathbf{A}$ are particular eigenvectors of $\mathbf{A}\mathbf{A}^T$ [86]. However, when no truncation is utilised (i.e. the number of modes is allowed to increase with every additional observation), this incremental procedure can also be too expensive since each step requires a batch SVD operation on a matrix the size of the current number of modes. As discussed in [20], incremental SVD yields significant computational savings only when the number of modes of $\mathbf{A}$ is kept at a number much smaller than the size of $\mathbf{A}$. To make the computation of the appearance covariance tractable for large problems, the number of appearance modes M_a must be chosen *a-priori*.

Extrinsic Appearance Variation

As with shape, to facilitate a compact intrinsic model of appearance, the effects of extrinsic (global) appearance variation should be accounted for separately. These extrinsic variations account for the different imaging (lighting) conditions under which the visual object is observed. The appearance of a visual object is then synthesised by composing the intrinsic and extrinsic appearance generating functions:

$$\mathscr{A}(\mathbf{p}_a, \mathbf{g}_a) \colon \Re^{M_s+G_a} \rightarrow \Re^{D_a P} = \mathscr{A}_g(\diamond; \mathbf{g}_a) \circ \mathscr{A}_l(\mathbf{p}_a), \tag{2.15}$$



(a) (b)

Figure 2.6: (a): Example of the first two modes of combined appearance variation for a human face learnt from the IMM Face database [89]. **(b):** Example of the first two modes of combined appearance variation for the left ventricle learnt from the database described in [112]. Each mode of variation is varied between ± 3 standard deviations, keeping the other parameters at zero. Note that the LDM instance used here is an AAM. As such, the model's triangulation is shown to illustrate the simultaneous variation in shape, along with appearance.

where $\mathscr{A}_g$ is the extrinsic lighting generating function, parameterised by $\mathbf{g}_a^{(G_a)}$.

The most common model of extrinsic appearance variation is the linear lighting model:

$$\mathscr{T}_g(\mathbf{a};\mathbf{g}_a)\colon \Re^{D_a P} \times \Re^2 \rightarrow \Re^{D_a P} = c\,\mathbf{a} + d\mathbf{1}^{(D_a P)}, \tag{2.16}$$

where $\mathbf{g}_a = [\ c\ ;\ d\ ]$, with c denoting the global lighting gain and d denoting the bias. Normalising the linear lighting effects over the training set involves an iterative process, similar to the generalised Procrustes alignment of shapes, where the cropped images are aligned, in the linear lighting model sense, to the mean image, and the mean appearance recomputed.

In the case of 3DMMs, a more accurate generative model of extrinsic lighting effects is utilised. The standard Phong [46] model is often chosen here, where the diffuse and specular reflections on a surface are approximately described. This involves a parameterisation of ambient light, the direction and intensity of directed light, specular reflectance of the object, and the angular distribution of specular reflections (see [101] for details).

2.1.3 Combined Appearance Parameterisation

In some cases it is beneficial to account for the correlations between the intrinsic shape and appearance of an LDM. This parameterisation, commonly used in AAMs, is denoted the *combined* appearance model, as opposed to the *independent* appearance model described previously. For visual objects exhibiting strong correlations between shape and texture, this repre-

sentation generally exhibits a more compact representation than its independent counterpart.

Using the intrinsic shape and appearance models described previously, the optimal parameters for every image in the training set can be obtained. The training set for the combined appearance model then consists of a concatenation of $\mathbf{p}_s$ and $\mathbf{p}_a$ into the vector $\mathbf{c} = [\, \mathbf{W}\,\mathbf{p}_s \,;\, \mathbf{p}_a \,]$, for each training image. Here, $\mathbf{W}$ is a diagonal scaling matrix, which accounts for differences between the units of measurement in shape and appearance. A common choice for $\mathbf{W}$ is an isotropic diagonal matrix where the diagonal entries are set to the ratio between the sum-squared eigenvalues of the independent shape and appearance models.

By applying PCA on these training vectors, a combined appearance model is obtained. New instances of the intrinsic shape and texture parameters can then be synthesised using: $\mathbf{c} = \mathbf{\Phi}_c\,\mathbf{p}_c$, where $\mathbf{\Phi}_c$ is the $((M_s + M_a) \times M_c)$ combined appearance basis matrix and $\mathbf{p}_c^{(M_c)}$ is a vector of combined appearance parameters. Note that the mean of the training data is zero, since the parameters are obtained from the application of PCA on the same training set, independently over the shape and appearance. The choice of M_c can be made using the same techniques as described in Section 2.1.1 for the shape model.

With this parameterisation, the linear shape and appearance in Equations (2.2) and (2.12) exhibit a change in their basis modes of variation:

$$\mathbf{\Phi}_s = \mathbf{\Phi}_s\mathbf{W}^{-1}\mathbf{\Phi}_{cs} \quad \text{and} \quad \mathbf{\Phi}_a = \mathbf{\Phi}_a\mathbf{\Phi}_{ca}, \quad \text{where} \quad \mathbf{\Phi}_c = \left[\mathbf{\Phi}_{cs}^{(M_s \times M_c)}; \mathbf{\Phi}_{ca}^{(M_a \times M_c)}\right]. \tag{2.17}$$

The linear shape and texture are now both driven by $\mathbf{p}_c$ rather than by $\mathbf{p}_s$ and $\mathbf{p}_a$ separately. Figure 2.6 illustrates the effects of varying the combined appearance parameters on the synthesised model's shape and appearance.

2.1.4 Other Representations

The method for modelling shape and appearance variability of deformable visual objects, described in the previous sections, is by far the most common due to its simplicity and compact representation. However, it is by no means the only approach. In this section, some other existing approaches are briefly discussed, along with their domain of application.

Sparse Linear Modelling

Although the variance maximising orthogonal bases for modes of appearance obtained by PCA are able to represent variability within an object class with a relatively small number of parameters, these modes of variation exhibit the characteristic that global deformations are preferred over local ones. This can compound the effects due to chance correlations between deformations inherent in a limited size training set. As many interesting characteristics of an object's variation are spatially localised (an example of this is a smiling face), an uncorrelated basis may be suboptimal for exploratory analysis. In light of this problem, some authors have proposed an alternative representation of an object's variability that directly favours locality.

An interesting method to apply here is the Independent Component Analysis (ICA). This method was used in [130] to represent statistical shape models. In [131] a comparison between ICA and PCA for MR cardiac segmentation using the AAM is presented. Pathology detection using an ICA based AAM is described in [116] and [117].

Another representation that favours locality can be obtained by applying an orthomax rotation to the principle components obtained through PCA as proposed in [115]. The result of applying this rotation to the uncorrelated bases is a *sparse* set of modes with strong local correlations. One of the advantage of this representation as compared to ICA or sparse PCA is that this rotation can be obtained for very high dimensional data such as appearance.

Nonlinear Modelling

Although the linear model class assumption works well in many applications, such as frontal faces and a number of medical image problems, in the case where a Gaussian distribution is a poor approximation of the true distribution of the object's shape and/or appearance, the following two problems result: (1) the model can reach invalid shape/appearance regions and (2) a lack of compactness can result. To tackle this problem, a number of authors have proposed nonlinear models to parameterise deformable visual objects.

LDM flavours that exhibit only a small number of landmarks, such as the ASM and AAM, afford the application of powerful nonlinear modelling techniques. In [103], Kernel PCA (KPCA) [107] was utilised to account for the nonlinear variations in the shape of visual objects that exhibit large pose changes. The intention of using KPCA is to restrict the possible instantiations of the model to valid shapes on the object's shape manifold. Here, the valid shape region was defined by placing an upper bound on the modulus of each of the normalised KPCA components in a similar manner to linear PCA, where the parameters are often bounded to lie within ± 3 standard deviations of the mean. In [129], it was argued that this method of restriction is invalid in the kernel space since the KPCA components do not behave in a similar manner to linear PCA components (i.e. zero KPCA components correspond to shapes far from the data and absolute values of all components are bounded). They then proposed restricting the KPCA parameters by placing a *lower* bound on the allowed 'proximity data measure', the distance from the origin in KPCA space. This is justified through the insight that the sub-manifold of the data is bounded and brackets the mean. In either case, the main difficulty of KPCA is that the construction of shapes from a set of KPCA parameters requires a nonlinear optimisation. Although affordable for shapes that exhibit a relatively small number of dimensions, their extrapolation to texture modelling is not generally viable due to its high dimensionality, often in excess of 10000 pixels.

Perhaps the simplest, albeit inelegant, solution to nonlinear appearance modelling is to partition the space into subspaces where linear approximations are reasonable. In [34] the nonlinear variations in shape and texture of a human face, brought upon by large in-plane pose changes, are tackled by partitioning the appearance space based on the pose of the face. A more principled partitioning scheme is presented in [26], where a Gaussian mixture model (GMM) is trained on a talking mouth sequences using Expectation Maximisation [14]. In order for the mixture membership evaluations to be computationally tractable, the GMM is defined over the space of PCA parameters of the whole set. Although this method avoids the reliance on heuristic parameters and partitioning such as pose, it still requires the number of partitionings to be set *a-priori*. Furthermore, it models only nonlinearities within the subspace defined by the PCA modes, restricting its representative capacity to the linear PCA model.

Despite the large literature on nonlinear distribution modelling and manifold learning, they

have rarely been implemented in the context of an LDM. As described above, the main difficulty is in modelling the appearance, which resides in a very high dimensional space. As most LDM applications are aimed towards alignment and tracking, online performance considerations become the most pressing issue, negating some of the benefits of improved representation accuracy afforded by nonlinear modelling.

2.2 The Automatic Learning of Correspondences

One of the main drawbacks of LDMs is that they require annotations, relating the same physical locations across the whole training set. Manually annotating large datasets is both tedious and error prone. Furthermore, when a dense shape model is used, such as in the 3DMM, manual annotations can only be made for a subset of the correspondences. Although most current applications that utilise LDMs still use hand labelled datasets, there have recently been advances in (semi)automatic techniques that have the potential to significantly ease the model building process.

The main aim of most automatic model building techniques is to find a set of corresponding landmarks in each image, which simultaneously accounts for the maximum amount of shape variation within the set and has minimal representation error over the training set. Compared to LDM fitting methods (see Section 2.3), automatic correspondence learning for LDM building is less explored. However, their approaches can be broadly categorised into two groups: feature based and image based.

Feature based approaches, for example [27; 60; 137], find correspondences between salient image structures (features), such as corners and edges in the image, by examining the local structure of the features. Once detected, the set of candidate features is matched across the whole image set, possibly utilising a geometric consistency criterion. The advantage of this approach is that feature comparisons and calculations are relatively cheap. The downside, however, is twofold. Firstly, there may be insufficient salient features in the object to build a good appearance model. Secondly, as the feature comparisons generally consider only local image structure, the global image structure on which the LDM is then modelled, is ignored. As a result, models built using annotations found in this manner may be suboptimal.

Image based approaches alleviate these problems by starting with the requirement of model compactness and faithful reconstruction. Most image based approaches utilise an image morphing and matching process in a group-wise fashion. Approaches of this kind typically learn the shape and appearance model of the LDM, along with the correspondences, by alternating solutions for the model whilst keeping the correspondences fixed, with solutions for the correspondences, whilst keeping the model fixed. Although this approach has no proof of convergence[2], the approach is fairly stable, affording numerous reports with encouraging results.

The pioneering method for the direct groupwise approach was presented in [133] for learning dense correspondences for use in a 3DMM. Utilising the current estimate of the model, the LDM is fitted to each image in the database. From locations defined by the fitted LDM's shape, optical flow is performed between the image and the LDM's appearance that is projected onto

[2]The direct groupwise method essentially solves a different problem at each cycle of the two step alternating procedure. As such, no common objective is maximised throughout the procedure.

the image frame. Using the landmarks, perturbed through the optical flow procedure, a new model of shape and appearance is built. This procedure is repeated a number of times, increasing the number of shape and appearance modes, until convergence is declared by examining the change in landmark perturbations between iterations. The main strength of this method is the simplicity with which correspondences can be obtained. The main drawback of the method is that it is prone to overestimating shape variations, since it relies only on the truncated SVD procedure, used in shape model building, to regularise the perturbed landmarks. The results reported by using this method were only evaluated qualitatively, based on the quality of the reconstructed appearance alone. Nonetheless, as the application domain of 3DMMs is often in computer graphics, this approach is still meritorious.

More recently, a number of methods have been proposed to address the drawback of the original method described above. In [9], Baker *et al.* do away with the two step procedure of model fitting and perturbation estimation by directly optimising the landmarks in all images, with the common objective of model reconstruction[3]. They also utilise a regularisation term in their objective, in order to promote smooth deformations in directions not yet accounted for by the current shape model. To achieve a reduced computational cost, the efficient project-out inverse compositional formulation [83] was used to minimise the objective. Although this method alleviates the overestimation of shape variability, compared to the method proposed in [133], a greater number of free parameters are required to be selected manually. However, it is suggested that since the method affords an efficient evaluation, a number of trials using different settings of the free parameters may still be possible. Finally, the applicability of this method for visual objects that exhibit large amounts of variability is yet to be verified[4]. In [54], it has been shown that the project-out inverse compositional method performs well only on visual objects with small amounts of shape and appearance variability.

In [33] a more powerful method is proposed, where the Minimum Description Length (MDL) of the whole training set is optimised. The method affords non-Gaussian distributions in shape and texture, although the results reported in this work utilise a linear model only. It also evaluates the model fit criterion in the image frame, rather than the more conventional model frame. This, they argue, alleviates the problem resulting from model frame evaluation, where the landmarks may distort to minimise the effects of hard-to-model regions in the image, resulting in erroneous correspondences. In any case, evaluation in the image frame is required here, due to the MDL criterion used. As such, the method requires an *inverse* warping procedure to project the model's appearance onto the image frame. This is a much more computationally demanding procedure than the forward warping procedure, even for the piecewise affine warp, utilised in this work. Furthermore, the gradients of the objective function must be evaluated using numerical differentiation techniques. To improve efficiency, a coarse-to-fine procedure is implemented, where the landmarks are perturbed through a *smooth* deformation field, controlled by a set of knots, placed at increasingly fine locations throughout the coarse-to-fine procedure. Finally, the same authors have published a number of other similar works, see [128] for example, which utilise essentially the same procedure, with a more theoretical

[3]Note that although a common objective is minimised at each step, it is done so in an alternating fashion, where the shape model, shape parameters, appearance model, appearance parameters, and the correspondences are each chosen to minimise the objective, keeping the others fixed whilst doing so.

[4]The experiments presented in [9] include a synthetic box and a person specific database.

treatment and customisable warping functions.

Although not designed specifically for LDMs, an interesting idea is presented in [63] that lends itself nicely to the problem of correspondence learning. Here, the orderings of vectorised images are optimised in order to maximise the likelihood of the data being generated by a Gaussian distribution. As the objective is convex, it obtains the globally optimal ordering of pixels for all images in the database. From these orderings, a dense correspondence set between images can be obtained. Choosing a set of these as landmarks, the correspondences required of LDM model building can be readily obtained. However, the Gaussian assumption may be a poor one, especially when other objects, apart from the visual object of interest, are present in the image. For example, clothing worn by subjects in a face database will not generally be Gaussian distributed. To tackle this problem, the same author extends the method to model nonlinear distributions in [64], where the aim now is to maximise the likelihood of a KPCA. Unfortunately, the resulting problem is nonlinear, affording only a locally optimal solution. The main drawback of utilising this method for LDM correspondence learning is that it effectively solves a Maximum Likelihood (ML) problem. No priors are placed in the deformation of the shapes between images, resulting in an overestimation of the shape variability.

The methods described above are representative of the currently existing methods for automatic LDM building. Although they exhibit reasonable performance on constrained databases, a solution for the general case is still an open problem. As such, in practice, the annotation process is still performed manually or using a semi-automatic approach [1].

2.3 Linear Deformable Model Fitting

LDM fitting is the process of finding the model parameters $\mathbf{p} = [\ \mathbf{p}_s\ ;\ \mathbf{g}_s\ ;\ \mathbf{p}_a\ ;\ \mathbf{g}_a\ ]$ that best describe the object in an image $\mathscr{I}$. Due to the nonlinearity of the problem, LDM fitting is usually achieved through an iterative process that sequentially updates the model parameters $\mathbf{p}$ through an update function:

$$\Delta\mathbf{p} = \mathscr{U}(\diamond;\mathbf{p}) \circ \mathscr{F}(\mathscr{I};\mathbf{p}), \tag{2.18}$$

where $\mathscr{F}$ is a feature extraction function that represents the image $\mathscr{I}$ from the perspective of the LDM at its current parameter settings, $\Delta\mathbf{p}$ are the updates to be applied to the current parameters and $\mathscr{U}$ is the update model that may depend on the current model parameters. A good coupling between $\mathscr{U}$ and $\mathscr{F}$ is generally required to ensure accurate predictions of the updates.

There exists a large variety of LDM fitting procedures, some of which are specialised to specific visual object categories, while others are tuned to specific applications. Despite the various approaches, all LDM fitting methods share the same five principle goals [101]:

- *Accuracy* : From an analytic perspective, the extraction of a fitted LDM's parameters is nothing more than a front end to the analysis of information contained in the image. As such, the accuracy of fitting is often vital to the utility of inference made using the LDM's parameters. Although uncertainty regarding fitting accuracy can be incorporated into the analysis, degradation of the results may be difficult to avoid, or underestimation may

result due to overregulaisation. This can be seen, for example, in the case of audio-visual speech recognition [88], where analysis of simple patch extracts outperforms analysis using AAM features due to fitting inaccuracies. On the other hand, highly accurate fitting may not be essential to the usefulness of LDMs in a particular application. Many graphical applications, avatar animation [49] for example, can still be implemented with aesthetically pleasing results despite achieving less than perfect fitting.

- *Efficiency*: Although better fitting efficiency is desirable in any application, in some cases it may be more important than others. When online processing is desired, such as in the online analysis of medical data [149], highly efficient fitting procedures are vital. On the other hand, in many problems for which LDM's can be utilised, for example photograph/video reanimation [17], efficiency, though desirable, is of less importance.

- *Robustness and Generalisability*: In many problems, the visual object of interest may exhibit large amounts of variability in its application domain compared to its available training data. As such, robustness to these unmodelled variations is highly desirable, and in many cases, vital in moving an application from development to production. However, there are cases in which the domain of the application is very constrained, where variabilities from the training data is minimal. Since incorporating robustness generally involves complexifying the fitting algorithm, it is sometimes desirable to deploy non-robust fitting procedures in these constrained cases.

- *Automatic behaviour* : Minimising or even eliminating user intervention in the fitting procedure is an important goal, especially in real time applications. However, due to the complexities involved in the fitting problem in general, some models may require user input in order to achieve good fitting accuracy [101].

- *Applicability* : A good fitting procedure should generalise well over a large domain of parameterisations. Although algorithms specific to a particular parameterisation may better fulfil some of the other goals described above, their applicability may be limited, and their contribution to the field in general, weak.

Fulfilling the five aforementioned goals is desirable in any fitting procedure. However, most methods favour the fulfilment of some of these goals over others. The choice regarding which goals to address is generally problem dependent, reflected by the numerous fitting algorithms in the literature. In the following sections, a discussion of the prevailing methods for LDM fitting is presented, in which the main goals that are (partially) fulfilled are identified in each.

2.3.1 The Search and Constrain Approach

One of the earliest methods for LDM fitting was proposed in conjunction with the first statistically based LDM, the ASM [31]. This method, which will be referred to as the "search and constrain" approach, combines the efficiency of local appearance matching with the regularising qualities of the LDM's statistical shape model. The general algorithm alternates the following two steps until convergence is achieved:

- For each landmark in the LDM's shape, find the perturbation from its current location that minimises the difference between the local appearance of that landmark and the image.

- Project the deformed landmarks onto the domain of plausible shapes, defined by the LDM's model of intrinsic and extrinsic shape variations, and regularise the shape parameters within this space.

This approach has similarities to the Demons algorithm [122], where the second step is replaced by a diffusion-like regularisation, projecting the current estimates onto the space of smooth deformations. The main strength of this approach is its efficiency, since calculating the deformations of each landmark is performed independently of all others, resulting in a problem with only a very small parameter set for each.

The ASM's search and constrain procedure utilises the profile derivative appearance model, learnt by cropping a set of pixels from the training images along the profile of each landmark, which is often set to be perpendicular to a predefined connectivity between them. As such, the ASM utilises a separate appearance model for each landmark that describes the local appearance of image derivatives along the predefined profiles. Furthermore, in modelling the linear appearance model, all modes of appearance variations M_a are kept, resulting in a full Gaussian model. During a search, optimal perturbation for each landmark is constrained to lie along the profiles of each landmark in the image frame. Consequently, the method utilises a semi-exhaustive search along the profile, typically at integer locations, 5-7 pixels along the profile in each direction, where the quality of a deformation for each landmark is evaluated using the *Mahalanobis distance* [78].

The projection step generally involves finding the LDM parameters that best fit the perturbed landmarks in the image frame, constrained by the regularisation imposed on the parameter space:

$$\mathscr{C}\left(\{\mathbf{x}\}_{i=1}^{n};\mathbf{p}_s,\mathbf{g}_s\right)=\mathscr{D}\left(\{\mathbf{x}\}_{i=1}^{n};\mathbf{p}_s,\mathbf{g}_s\right)+\lambda\mathscr{R}(\mathbf{p}_s), \tag{2.19}$$

where $\{\mathbf{x}_i\}_{i=1}^{N}$ are the perturbed landmark locations, $\mathscr{D}$ penalises the distance between the perturbed and projected model's landmarks (usually set to the least squares error), $\mathscr{R}$ regularises the intrinsic shape parameters by penalising complex deformations, and λ is a weighting factor. There are two common regularisers used here. The first is to assume the intrinsic parameters are Gaussian distributed:

$$\mathscr{R}_s(\mathbf{p}_s)=\mathbf{p}_s^T\boldsymbol{\Sigma}_s^{-1}\mathbf{p}_s, \tag{2.20}$$

where $\boldsymbol{\Sigma}_s$ is the covariance matrix of the shape. The advantage of this regularisation is that the problem to be solved becomes one of MAP (Maximum *a-posteriori*) estimation. The use of a Gaussian prior on the object's shape has been successfully utilised in a number of works, for example [101]. The second type of regularisation is generally implemented by constraining the intrinsic shape parameters to their feasible domain, generally defined as a box constraint within $\pm L$ standard deviations, or as a hyperellipsoid constraint defined by:

$$\mathbf{p}_s^T\boldsymbol{\Sigma}_s^{-1}\mathbf{p}_s=L^2. \tag{2.21}$$

This is equivalent to assuming a uniform prior within the feasible domain, with zero probability everywhere else. The advantage of this regulariser is that optimisation can be performed by alternately solving the data term and constraining the solution, each of which affords an efficient evaluation. The drawback of this approach is that it effectively performs a ML (maximum likelihood) estimation that may be less robust than a MAP estimate.

Although the search and constrain approach for ASM fitting exhibits good efficiency, its domain of application is limited. It requires that the visual object of interest exhibits a large number of strong edges. Furthermore, fitting accuracy and robustness to initialisation conditions for this implementation are limited. One of the main culprits of this drawback is the limited search domain for the landmark perturbations, which are constrained to lie along the profile of each landmark, relying on the projection process to regularise the parameter updates in such a way that the fitting error is reduced. Recently, the domain of the ASM's search step has been extended to utilise a 2D region around each current landmark. In [35], a boosted classifier, utilising the efficient Haar-like features was used to rapidly evaluate the fitness of landmark perturbations. A logistic regressor was also proposed to predict landmark perturbations, a process that avoids an exhaustive local search.

The utility of the search and constrain approach has also be demonstrated in more complex 3D models. In [102], a 3DMM was fitted to an image by sequentially estimating an optical flow field between the projected model's texture and the image, and using the destination of the flow as the perturbed landmarks in Equation (2.19). The method also achieves significant computational savings over other 3DMM fitting approaches, by virtue of the bilinear relationship between the perturbed landmarks and the shape and rigid parameters in the constrain step. One drawback of this approach, which is typical of most search and constrain approaches, is that each landmark is given equal weighting in the constrain step. In [139], normalised cross-correlation between patches in a template image were matched to the image, requiring only one training image (though a statistical shape model is still required), where the projection process is regularised by the correlation score. As such, the projection process favours fitting landmark perturbations that exhibit similar appearance to the template. A more formal treatment of this problem is presented in [13], albeit in a tracking scenario. Here, fitting does not rely on an appearance model. Instead, the perturbations are obtained through optical flow estimates, the anisotropic uncertainty of which is directly incorporated into the objective of the constraining step. By utilising a non-spherical error norm, the information state of the system is maximised, resulting in a better inference of the desired shape parameters.

2.3.2 Generative Fitting

Generative fitting methods pose fitting as the minimisation/maximisation of some measure of fitness between the LDM's synthesised appearance with that of the warped image, with an optional regularisation over the parameters:

$$\mathscr{C}(\mathscr{I}; \mathbf{p}_s, \mathbf{g}_s, \mathbf{p}_a, \mathbf{g}_a) = \mathscr{D}(\mathscr{I}; \mathbf{p}_s, \mathbf{g}_s, \mathbf{p}_a, \mathbf{g}_a) + \lambda_s \mathscr{R}_s(\mathbf{p}_s) + \lambda_a \mathscr{R}_a(\mathbf{p}_a), \tag{2.22}$$

where $\mathscr{D}$ is the fitness function, $\mathscr{R}_s$ and $\mathscr{R}_a$ are regularisation functions over the intrinsic shape and appearance parameters, respectively, and $\{\lambda_s, \lambda_a\}$ are design parameters that trade-off the contribution of the image fitness and parameter regularisation. An illustration of the fitting

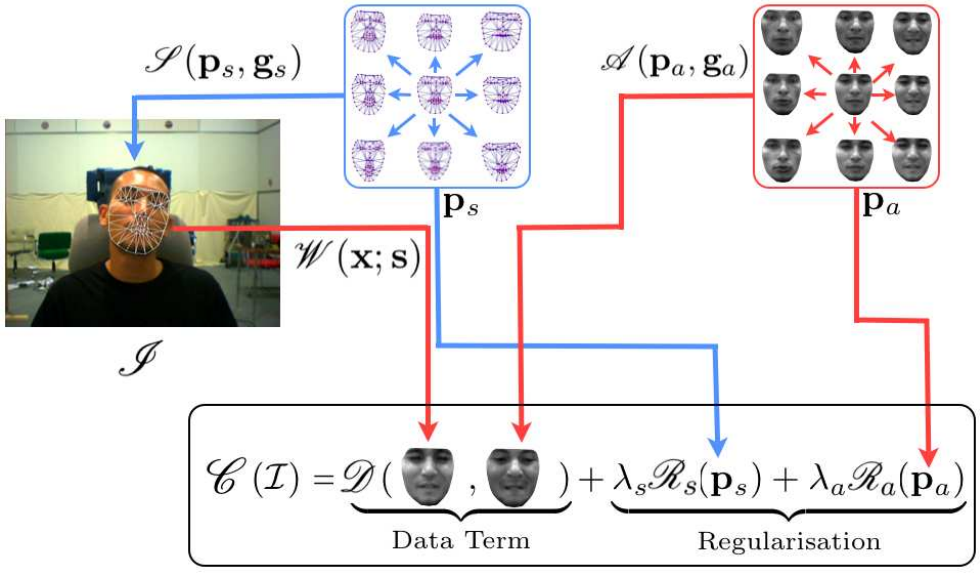


Figure 2.7: An illustration of generative LDM Fitting.

process and its various components is presented in Figure 2.7.

By far the most common fitness function used in LDM fitting is the least squares error, or a robust variant thereof:

$$\mathscr{D}(\mathscr{I}; \mathbf{p}_s, \mathbf{g}_s, \mathbf{p}_a, \mathbf{g}_a) = \sum_{\mathbf{x} \in \mathbf{\Omega}} \rho \left(\left[\mathscr{A}(\mathbf{x}; \mathbf{p}_a, \mathbf{g}_a) - \mathscr{I} \circ \mathscr{W}(\mathbf{x}; \mathbf{p}_s, \mathbf{g}_s) \right]^2 ; \sigma \right), \tag{2.23}$$

where $\mathbf{\Omega}$ is the spatial domain in the canonical frame, over which the LDM's appearance is defined. The function ρ is usually taken either as the identity, in which case the problem is in least squares form [12; 30; 82], or a robust function, in which case σ denotes a sensitivity parameter and the cost function is an iteratively reweighted least squares problem [119]. As such, the Gauss-Newton method is an attractive optimisation method to implement here, as it requires only first order derivatives and its convergence properties are well understood [94]. The Lucas-Kanade approach [76], which was implemented in the context of general image alignment in [8], is essentially a Gauss-Newton optimisation, specialised to the case of ML image alignment. The main drawback of the Lucas-Kanade method is that it is extremely inefficient. Despite requiring only first order derivatives, due to the relatively large number of parameters in an LDM, compared to rigid image alignment, these derivative computations, along with the Gauss-Newton Hessian and its inversion, are computationally expensive. Referring to the update form in Equation (2.18), the update model $\mathscr{U}$ relies on the current parameters $\mathbf{p}$. As such, most generative approaches to AAM fitting either assume some parts of the update computation are fixed or reformulate the problem such that they are.

One of the earliest methods for generative fitting that achieved reasonable fitting efficiency was proposed in the context of AAMs in [30]. In this work, Cootes *et al.* assume that the Jacobian of the least squares problem in Equation (2.23) is fixed. This results in a linear update model that can be precomputed, allowing rapid fitting to be achieved. In training, the Jacobian

is obtained by averaging a number of Jacobian estimates in every image, each estimated using numerical differentiation. Since the fixed Jacobian assumption holds only approximately, the method utilises a simple step size adaptation procedure, whereby the estimated updates are sequentially halved until a reduction in appearance reconstruction error is achieved. More recently, in a method coined adaptive AAM [12], the fixed Jacobian assumption is relaxed by decomposing it and assuming only the component pertaining to the derivative of the warping function is fixed. The resulting method exhibits improved accuracy, however, as the linear update model depends on the current appearance parameters, the fitting procedure is still computationally expensive.

Adaptations of the inverse-compositional image alignment [7] to LDM fitting have also gained momentum recently. By reversing the roles of the image and the model in the error function, the derivative of the warping function is fixed, resulting in significant computational savings. The project-out method [82], for example, minimises the cost in a subspace orthogonal to the modes of appearance variation, resulting in an analytically fixed linear update model. Despite exhibiting one of the most computationally efficient fitting procedure to date, it works well only for objects exhibiting small amounts of variabilities [54]. This problem is partially addressed by the simultaneous method [4], which solves for the shape and appearance parameters simultaneously. However, similar to the method in [12], its update model depends on the current appearance parameters, again resulting in a computationally expensive fitting procedure. Recently, the simultaneous inverse compositional method has been adapted for use in the 3DMM [104]. This requires a reformulation of the composition operation to be applicable to 3D shape models. Furthermore, by virtue of the separate treatment of image and model frames, they are able to utilise a fixed Jacobian, allowing the linear update model to be precomputed.

In real world problems, the visual object in images that are to be fitted by an LDM often exhibit unmodelled appearance variations. This may be caused, for example, by occlusions. There have been various attempts to robustify LDM fitting against these gross appearance differences. One such approach was presented in [97], where the multimodal nature of the reconstruction error histogram is analysed in order to distinguish outliers from inliers. Analysis, here, involves a selection procedure, where the effects of including particular histogram modes are assessed through their impact on the matching procedure. Although this procedure has been shown to accurately distinguish inliers from outliers, the involved procedure can be computationally demanding, leading to inefficient fitting. Most other attempts to robustify LDM fitting simply replace the least squares appearance fitting criterion with a robust one. This approach was taken in [55], where the approach was implemented within the inverse-compositional framework to promote efficiency. However, since the parameter update model relies on the current robust scalings, it cannot be precomputed. To reduce computational complexity further, the authors assume the outlying pixels exhibit a degree of spatial coherence by utilising the same robust weight for all pixels within a predefined regions. This results in an extremely efficient fitting procedure for person specific cases, achieving real time fitting in their experiments. An alternative to utilising spatial coherence was proposed in [104], where pixels exhibiting errors larger than a predefined threshold are simply taken out of the optimisation procedure, at each iteration. This way, the Jacobian is fixed, allowing efficient fitting to be achieved. The problem with this approach is that large errors during fitting do not only correspond to outlying pixels, but also to inlying pixels that are misaligned at the current state

of optimisation. As such, excluding all pixels with large errors from the fitting procedure will generally underestimate the parameter update step, leading to slow convergence. More recently in [99], the efficiency penalty stemming from robustifying the fitting criterion was addressed by applying the effects of outliers directly on the appearance residual vector. As such, the pre-computed non-robust update model can be used, allowing efficient evaluation. However, the procedure is only an approximation of applying the weighting procedure to the Hessian and gradient of the iteratively reweighted least squares problem, leading to parameter updates that are biased in favour of the identified inliers. This is partially addressed in that work, through a type of deterministic annealing procedure, where two sets of robust scaling parameters are chosen to account for errors in the early and later stages of fitting.

Finally, although most generative approaches utilise a variant of the least squares error, there are a small number of methods that venture away from this norm. In particular, the method in [75] adapts the support vector tracking method [3], to the case of AAMs. Here, the objective is defined as a support vector machine classification score. To achieve efficient evaluation, Haar-like features, the gradients of which are fixed, are used to process the image. The main drawback of this approach is that the support vector machine classification score can exhibit significant amounts of local minima. This difficulty with the support vector tracking framework was investigated in [141], where it was found that due to the highly nonlinear fitting criterion, a discriminative approach was capable of achieving better estimates, using the relevance vector machine [123] to predict the updates.

2.3.3 Discriminative Fitting

The discriminative approach to LDM fitting directly learns a fixed linear relationship between the features $\mathscr{F}(\mathscr{I};\mathbf{p})$ and the parameter updates $\Delta\mathbf{p}$, given a training set of perturbed model parameters:

$$\{\mathscr{F}(\mathscr{I}_i;\mathbf{p}_i^*-\Delta\mathbf{p}_i)\ ,\ \Delta\mathbf{p}\}_{i=1}^{N_d}\,, \tag{2.24}$$

where $\mathbf{p}^*$ is the optimal parameter setting for the i^{th} sample and N_d is the total number of perturbations in the training set. The main advantage of this approach is its efficiency, since $\mathscr{U}$ can be prelearnt. Compared to methods for generative LDM fitting, there are significantly fewer existing methods that utilise discriminative LDM fitting. The main methods for discriminative fitting will be discussed below.

In the original AAM formulation [43], the linear update model was shown to approximately explain the relationship between the AAM's normalised appearance residual feature:

$$\mathscr{F}(\mathscr{I};\mathbf{p})=\bar{\mathbf{t}}+\mathbf{\Phi}_t\mathbf{p}_t-\mathscr{N}\circ\mathscr{I}\circ\mathscr{W}(\mathbf{p}) \tag{2.25}$$

and the parameter updates $\Delta\mathbf{p}$, around the optimal parameter settings $\mathbf{p}^*$ for a given image. $\mathscr{N}$ normalises the warped image so as to exhibit similar global lighting gain and bias as the model's texture. The update $\mathscr{U}$ is easily found through linear regression on the data set in Equation (2.24). Although this method was later superseded by the fixed Jacobian method [30], proposed by the same authors, it serves as an interesting first attempt to apply a discriminative approach to LDM fitting.

Since the original AAM formulation, research on the discriminative approach to LDM

fitting has focused mainly on the choice of $\mathscr{F}$ that better adheres to a linear relationship with the parameter updates. The direct appearance model method [62], for example, uses the PCA reduced appearance residuals and predicts the shape directly from the appearance. This method boasts significant memory savings in AAM training as well as improved fitting performance. In [40], a linear relationship is learnt between the canonical projections of the texture residuals and parameter updates. The method utilises canonical correlation analysis to find the subspaces that best adhere to a linear relationship. These methods have been shown to exhibit faster convergence and better accuracy compared to the original formulation in [43].

Another direction of research involving discriminative fitting is to investigate the utility of more sophisticated regressors in predicting parameter updates. In [141], the problem of template alignment and tracking was tackled from a discriminative perspective, utilising the relevance vector machine [123] to regress parameter updates. The nonlinear decision function afforded by this approach results in highly accurate fitting, outperforming the generative support vector tracking approach as discussed in the previous section. However, this method has yet to be adapted to the problem of LDM fitting. One of the difficulties in doing so is to do with the type of regressor used, where the kernel functions are evaluated using the raw image feature, which can be computationally expensive to evaluate, despite the sparsity of the relevance vector machine. In the case of template matching, at most six parameters need to be regressed, such as in the case of affine deformations. In LDM fitting, the model parameters typically range between 50 and 100. Furthermore, the feature vector used in LDMs generally exhibit a larger dimensionality, typically in the order of 10000, compared to the (20×20)-window used in template matching.

Recently in [150], the computational complexities involved in utilising a nonlinear regressor in discriminative fitting, were addressed through the utility of a boosted set of weak learners, which are based on the Haar-like features. Although a large number of weak learners need to be evaluated to regress the updates, typically in the order of 200 for each parameter, since the Haar-like features afford efficient evaluations through the use of the integral image [72], the estimation procedure was shown to achieve high efficiency. This method was extended in [144; 146] to account for prediction inaccuracies by performing the same estimation from a number of different locations around the initial parameter settings. A generative inference was then made to select the most likely configuration, based on predictions from the various perturbed initial settings. Also, a more sophisticated weak learner set was used in these works, in comparison to the original method in [150], which utilised a multimodal decision function.

2.4 Conclusion

In this chapter, a detailed discussion of the LDM has been presented. The main mathematical tools involved in LDM parameterisation were discussed, relating the various LDM flavours through a common nomenclature. A short overview of some of the less popular parameterisations was also presented. The problem of automatic model building was then discussed, highlighting the strengths and weaknesses of the various existing approaches. Finally, an overview of the various existing LDM fitting procedures was presented, where the methods were partitioned into three groups based on the principle strategy utilised in each.

Apart from the issues pertaining to LDMs discussed in this chapter, there exist a number

of other important aspects that would benefit from research. One of these is the issue of appearance representation. It has been shown in a number of works, for example [65; 68], that the representation of the LDM's appearance has significant effects on the performance of the LDM. However, current research into this aspect entails a 'hit-and-miss' approach, whereby different representations are evaluated without any real indication of optimality. A notable exception is that presented in [37], where the optimal filtering operation is selected to optimally smooth the fitting error terrain. Another aspect of LDMs that is often ignored in the literature is the effect of LDM fitting performance on the various applications for which it is intended. For example, the utility of LDMs for face recognition [42; 44], has been evaluated by directly observing the accuracy of the results. Only a limited amount of research has been done on investigating the effects of LDM fitting accuracy on the performance of the recogniser. This is a more difficult problem, however, since it requires a simultaneous treatment of LDM fitting, its representative power, and the recognition procedure that utilises it.

As a final note, with increasing computational power and theoretical advancement, the utility of nonlinear methods for representing the shape and appearance of visual objects may be affordable in the near future. This can be expected to improve the representative power even further over that afforded by the LDM. As such, it remains important that procedures, developed now for LDMs, exhibit sufficient flexibility, such that their adaptation to more sophisticated models can be achieved.

The Pairwise Learning of Correspondences

I miss you, but I haven't met you yet.

Björk

In order to build the shape and appearance models of an LDM, a set of homologous correspondences across a training set of images must first be available. Although many approaches for rigid object correspondences are now available (see [25; 70; 90], for example), solutions for the non-rigid case are still limited. The main difficulty lies in accounting for deformations exhibited by the visual object, where rigid geometric constraints, such as the structure tensor [2; 109], are not applicable. This matter is made worse by the inherent variations in appearance exhibited by deformable visual objects, making feature based approaches unreliable in all but the simplest of cases. As such, direct (generative) based approaches must be utilised to account for the spatial dependencies of the object's appearance.

In this chapter, a direct method for automatic correspondence learning between pairs of images is presented. Given a template image, for which a set of manually selected annotations is available, the aim of correspondence learning is to perform nonrigid registration between the template and all other images in the training set, such that correspondences over the whole set of images can be defined, allowing the statistical models of shape and appearance to be built. It relies on the assumption that the shape and appearance deformations in a visual object between a pair of images are (piecewise) *smooth*. By virtue of its Bayesian framework, all the free variables of the problem can be tuned automatically.

A formal description of the correspondence problem is presented in Section 3.1. Section 3.2 then outlines the generic Bayesian framework that is utilised in the pairwise learning procedure, with explications on the densities defining the problem given in Section 3.3. An approach for solving the Bayesian inference over the correspondences and parameterisations of the densities is presented in Sections 3.4 and 3.5. The capacity of the pairwise approach to provide meaningful correspondences is evaluated on the human face in Section 3.6, where experiments utilising person specific, pose specific and generic person databases are presented. Section 3.7 concludes this chapter with a general discussion and mention of directions of future work.

3.1 Problem Statement

Homologous correspondence denotes semantically equivalent locations in different instantiations of a visual object (see Figure 2.1 for an illustration). In manual annotations, this is often interpreted aesthetically as corresponding locations that are physically meaningful. As such, subjectivity plays a large role when correspondences are obtained manually, leading to biased annotations. Furthermore, the subjectivity induced correspondence errors do not, in general, follow an isotropic Gaussian distribution. This is best illustrated by points on an edge, where the well known aperture problem can lead different human experts to choose annotations at different locations along the edge. If the measurement noise is treated as being isotropically Gaussian distributed, using truncated SVD for example, will lead to biased correspondences. To make matters worse, some visual objects exhibit visual features that are present in some instances but not in others, for example a moustache on the human face. Although in this case, a human expert generally makes an annotation decision globally over all locations, relating the geometries of the object's instances, these considerations are still subjective and prone to differences in interpretation.

In automatic correspondence learning, photometric and geometric similarities constitute the measure of homology. Photometric similarity encapsulates the intuition that corresponding points exhibit less appearance differences than non-corresponding points. The intuition here can be somewhat misleading, however, since there may exist instantiations of the visual object where corresponding points exhibit the same, if not more, appearance differences than some non-corresponding points. This is because photometric similarity is inherently a local descriptor that considers correspondences on a per-location basis. Using photometric similarity alone, therefore, will lead to spurious correspondences. Geometric similarities, on the other hand, are implemented in such a way as to encapsulate the intuition about the *topological* rigidity of a visual object. As such, they are a global constraint on the correspondences, enforcing topological equivalence amongst the different instantiations of the visual object. Combined, the photometric and geometric similarities can be formulated in the well established regularised data fitting framework [61]:

$$\mathscr{C}\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\,;\,\mathbf{s}\right) = \mathscr{D}\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\,;\,\mathbf{s}\right) + \lambda\,\mathscr{R}\left(\mathbf{s}^0\,;\,\mathbf{s}\right), \tag{3.1}$$

where photometric similarities make up the data term $\mathscr{D}$ and the geometric similarities make up the regularisation term $\mathscr{R}$, with the trade-off between them weighted λ. Here, $\left\{\mathscr{I}^0, \mathbf{s}^0\right\}$ denote the template image and its annotation respectively. $\mathscr{I}$ denotes an un-annotated image that contains an instance of the visual object of interest, possibly taken under different imaging conditions, and $\mathbf{s}$ denotes the n locations in $\mathscr{I}$ that correspond to $\mathbf{s}^0$ in $\mathscr{I}^0$:

$$\mathbf{s} = [\, x_1\,;\, y_1\,;\, \ldots\,;\, x_n\,;\, y_n\,]. \tag{3.2}$$

In discussions that follow, the correspondence set of a particular image will also be denoted as the "shape" of an image.

This thesis deals with correspondences in a *pseudo-dense* sense. A pseudo-dense set of correspondences is defined as a correspondence set for which manual annotations are still possible, but are generally impractical for large databases. Examples of LDMs that use a pseudo-dense

correspondence set include the ASM and AAM. In these models, the correspondences are often called landmarks as they often correspond to physically salient features, such as eye and mouth corners in the class of human faces. For most models of this kind, the number of landmarks range between 50 and 100. Compare this to a sparse annotation of two to four features, often required by generic structure recovery procedures, such as face recognition systems.

Suitable forms for the data and regularisation terms in Equation (3.1) are problem dependent as they rely on the underlying deformation structure of the visual object as well as the measurement noise of the images. A typical approach, when no other domain knowledge is available, is to assume the true correspondence set is that which satisfies the photometric constraints with the minimum amount of distortion, both in the object's shape and appearance. Here, distortion is generally chosen to measure irregularity of the deformations. This relates to the idea that points that are topologically close on the visual object vary in similar ways.

Regardless of how regularisation is formulated, the cost function in Equation (3.1) is almost always nonlinear due to the nonlinear relationship between the image intensities and the shape. As the visual object of interest can be located anywhere within an image, a sufficiently good initial estimate of the shapes in each image must be available in order for the optimisation procedure to have a reasonable chance of finding the global minimum or at least a good local one. In this chapter, it is assumed that a coarse estimate of the location and scale of the visual object in each image is available. This is typical for many visual databases for computer vision problems, where images are taken under known conditions, with the object roughly placed at the centre of the image at a particular distance. In the more general setting, a detector for the object class of interest may be available to provide this coarse level of information. Otherwise, manual annotation of the location and scale may be required. It should be noted that this type of annotation is relatively simple, since only a bounding box is required to obtain a coarse estimate of location and scale. In Figure 3.1, some images from a typical training set are shown along with their bounding box, which were found automatically using a publicly available face detector[1].

Finally, the regularisation weight λ in Equation (3.1) is generally unknown *a-priori*. A suitable choice for λ is often considered a 'black art' as it involves intuition about the problem as well as a trial and error procedure to refine the initial estimate. When only one regularisation weight is present in the cost function, a semi-exhaustive search for the optimal weight may be possible, evaluating the results for a number of different settings. However, the problem remains on how to evaluate the quality of a chosen setting. If ground truth data is available for more than one image, a cross validation procedure can be utilised to find the best weight. This approach is still problematic however, since the optimal weight for a given image will depend on the amount of deformation exhibited by the visual object in the image. As such, if the deformations in a test image exhibit magnitudes that are markedly different from those in the ground truth images, the best regularisation weight obtained from cross validation will be sub-optimal. This problem is complicated further when more than one regularisation weight is involved in order to account separately for different sources of deformations.

[1] `http://sourceforge.net/projects/opencvlibrary`

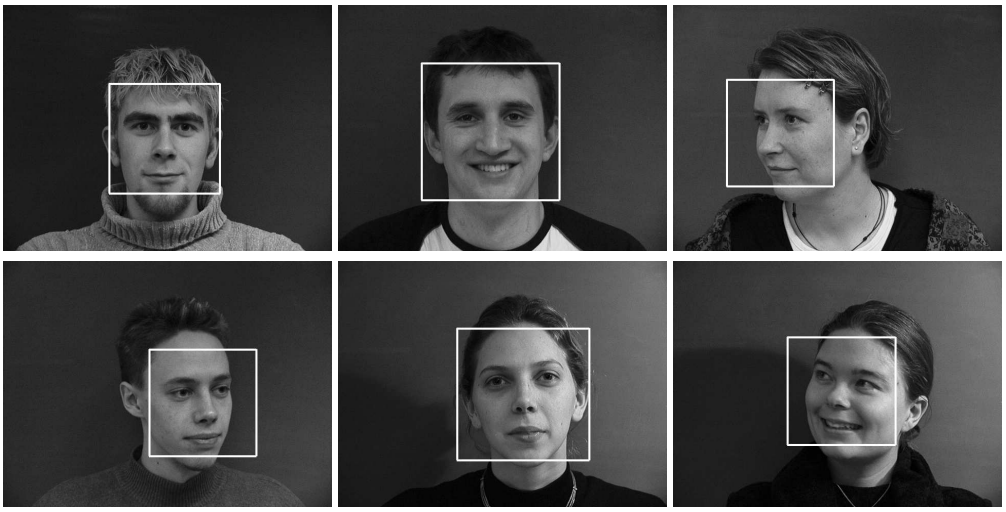


Figure 3.1: Some example images from the IMM Face database [89] with their automatically detected bounding box.

3.2 A Bayesian Framework

From a Bayesian perspective, the aim of pairwise correspondence learning is to maximise the likelihood of the shape in an image, given the pre-annotated template. Formally, the problem's objective is to maximise:

$$\begin{aligned} p\left(\mathbf{s} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\right) &= \int p\left(\mathbf{s}, \boldsymbol{\theta} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\right) d\boldsymbol{\theta} \\ &= \int p\left(\mathbf{s} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \boldsymbol{\theta}\right) p\left(\boldsymbol{\theta} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\right) d\boldsymbol{\theta}. \end{aligned} \tag{3.3}$$

Here, $\boldsymbol{\theta}$ denotes both the type and parameterisation of the probability density functions (PDFs) describing the distribution of the correspondences. As such, the (joint) Bayesian inference estimation process essentially integrates over all possible shape posterior densities, with each weighted by the model's likelihood given the data $\left\{\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\right\}$. However, integrating over all possible types of PDFs is not possible in general. As such, a sub-family of densities is usually chosen, with the integration performed over the parameterisation of that density alone. It should be noted here that the likelihood of each image in the database is assumed to be parameterised separately, independent of the density parameterisation of all other images. Furthermore, these likelihoods are also assumed to be dependent only on the shape for their respective image. By virtue of the separate parameterisations, this formulation has the ability to account for the variations in deformation magnitudes and image noise within the training set. In other words, the model is specialised to each image in the database separately.

The second term in the right most part of Equation (3.3) is the prior over the PDF's parameterisation, which can be decomposed as follows:

$$p\left(\boldsymbol{\theta} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\right) \propto p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0 \mid \boldsymbol{\theta}\right)\ p(\boldsymbol{\theta}). \tag{3.4}$$

In practice, the data likelihood can be found through a marginalisation over the shape:

$$p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0 \mid \boldsymbol{\theta}\right) = \int p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{s} \mid \boldsymbol{\theta}\right) d\mathbf{s} \propto \int p\left(\mathscr{I} \mid \mathbf{s}, \mathbf{s}^0, \mathscr{I}^0, \boldsymbol{\theta}\right)\ p\left(\mathbf{s} \mid \mathbf{s}^0, \boldsymbol{\theta}\right) d\mathbf{s}. \tag{3.5}$$

Furthermore, when no other information regarding the parameterisation is available, as is often the case for automatic correspondence learning, $p(\boldsymbol{\theta})$ is commonly assumed to be a non-informative (uniform) prior, leading to the marginalised maximum likelihood (MML) estimate. Substituting this form into Equation (3.3), we note that even for the case where $\boldsymbol{\theta}$ denotes parameterisations exclusively, the resulting form is generally quite complex, the analytic integration of which is rarely tractable. Therefore, Equation (3.3) is often approximated by [41; 84]:

$$p\left(\mathbf{s} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\right) \approx p\left(\mathscr{I} \mid \mathbf{s}, \mathbf{s}^0, \mathscr{I}^0, \boldsymbol{\theta}^*\right) p\left(\mathbf{s} \mid \mathbf{s}^0, \boldsymbol{\theta}^*\right) \text{ where } \boldsymbol{\theta}^* = \max_{\boldsymbol{\theta}} p\left(\boldsymbol{\theta} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\right). \tag{3.6}$$

In other words, rather than integrating over all possible densities, it is simpler to maximise inference over the density that maximises the likelihood over the data. Although this approximation is not connected formally with the formulation in Equation (3.3), this open gap between a very general theoretical formulation and a more humble practical approach is a pitfall widely observed in the literature. The effects of this approximation rely on the nature of the densities themselves and as such are problem dependent. In the case where $p\left(\boldsymbol{\theta} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\right)$ is a delta function, Equation (3.6) is no longer an approximation, however this will not generally be the case. It should be noted here that the aim of this section is not to bridge the gap between the true and approximate formulations, but rather to pose the automatic correspondence learning problem in a formal Bayesian framework and to point out that most currently existing approaches to solving this problem can be derived from this framework, each with their own approximations.

Using the approximate formulation in Equation (3.6), pairwise correspondence learning can be achieved by first calculating $\boldsymbol{\theta}$ that maximises Equation (3.4), fixing the densities in Equation (3.6) using $\boldsymbol{\theta}$ that was found previously, and finally maximising it with respect to the shape. Contrast this with the typical approach for automatic correspondence learning, where the maximum of Equation (3.6) is sought for a number of different settings of $\boldsymbol{\theta}$, choosing the correspondence set that optimises some heuristic measure of quality, which in many cases simply involves a subjective decision on that which qualitatively *looks* the best.

In practice, however, there exist some difficulties with this framework. In most cases, the likelihood of the images is defined by some measure of fit between an appearance model and the *warped* image, defined through the correspondences with which the relationship is generally nonlinear. As such, regardless of how the prior over the correspondences are defined, the marginalisation in Equation (3.5) will not result in an analytically integratable form for many interesting families of densities. Therefore, an approximation must be made here, whereby the likelihood density, in particular its components pertaining to the warped image, is approximated by a simpler form that affords an analytic solution to the integration. With this approximation, automatic correspondence learning, then, involves an iterative procedure that interleaves estimates of $\boldsymbol{\theta}^*$ and $\mathbf{s}$, improving the approximation of the true density in the

Algorithm 1 Generic Pairwise Learning of Correspondences: A MML/MAP Approach

Require: $\{\mathscr{I}^0, \mathbf{s}^0\}$ (template), $\mathscr{I}$ (image), $\mathbf{s}$ (initial correspondence estimate), $\boldsymbol{\theta}$ (initial parameter estimate)

1: **while** !converged $\{\mathbf{s}\}$ **do**
2: Approximate $p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{s} \mid \boldsymbol{\theta}\right)$ with a form that affords analytic integration.
3: Optimise: $\boldsymbol{\theta}^* = \max_{\boldsymbol{\theta}} p\left(\boldsymbol{\theta} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\right)$.
4: Optimise: $p\left(\mathbf{s} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0\right) \approx p\left(\mathscr{I} \mid \mathbf{s}, \mathbf{s}^0, \mathscr{I}^0, \boldsymbol{\theta}^*\right) p\left(\mathbf{s} \mid \mathbf{s}^0, \boldsymbol{\theta}^*\right)$.
5: **end while**
6: **return** $\mathbf{s}$

marginalisation each time. A summary of this procedure is outlined in Algorithm 1.

The formulation presented above is also known in the statistical community as the method of "hierarchical priors" [50]. Similar formulations have previously been utilised in [66] for optical flow estimation and in [151] for the problem of image completion, though in that work the integration is performed through a Monte Carlo simulation. In these works, the method is coined the combined *marginalised maximum likelihood/maximum a posteriori* (MML/MAP) estimator in reference to how the density parameters $\boldsymbol{\theta}$ (also called "hyperparameters") and the shape $\mathbf{s}$ are respectively estimated. An instantiation of this approach for solving the pairwise correspondence learning problem requires a number of interrelated components to be explicated, namely:

- The specific parameterisation of the visual object's deformations that define the correspondences.
- The densities describing the deformations as well as their approximations that allow tractable solutions to be attained.
- Optimisation procedures for steps 3 and 4 or Algorithm 1.

These three components will be dealt with in detail in the following sections.

3.3 Defining the Densities

There is a large body of research on pairwise nonrigid registration, especially in the domain of medical image analysis (see [71; 152] for surveys). However, there is a lack of consensus on a number of important aspects of the problem. Amongst others, variations in the different methods include the parameterisation of deformations, the measure of photometric similarity and the type of regularisation used. Although the choices for some of these components are problem dependent, in this section, prototypes for the conditional distribution models governing an image's shape and appearance, given that of the template, are presented, encoding domain knowledge regarding deformation smoothness. It should be noted that other prototypes may also be posed within the Bayesian framework presented in Section 3.2.

Examining Equations (3.5) and (3.6), the two densities that need to be defined for a particular specialisation of the MML/MAP method are the image likelihood $p\left(\mathscr{I} \mid \mathbf{s}, \mathscr{I}^0, \mathbf{s}^0, \boldsymbol{\theta}\right)$ and the prior over the shape $p\left(\mathbf{s} \mid \mathbf{s}^0, \boldsymbol{\theta}\right)$. Here, specialisation involves defining the family of

densities used to describe them as well as the parameterisation of the chosen family of densities. Once the forms of these densities have been defined (remember that $\boldsymbol{\theta}$ is assumed to only define the parameterisation of a chosen family of densities), optimisation strategies for both the model and correspondences can be developed using the MML/MAP's alternating two step procedure.

3.3.1 Defining the Likelihood

In Equation (3.6), $p\left(\mathscr{I} \mid \mathbf{s}, \mathbf{s}^0, \mathscr{I}^0, \boldsymbol{\theta}^*\right)$ denotes the likelihood of an image for a chosen shape and parameterisation. This quantity, which is maximised in a maximum likelihood problem, corresponds to the data term in the Equation (3.1). In the simplest case, this data term takes the form of a least squares problem:

$$\sum_{i=1}^{P} \left[\mathscr{I}^0 \circ \mathscr{W}\left(\mathbf{x}_i; \mathbf{s}^0\right) - \mathscr{I} \circ \mathscr{W}(\mathbf{x}_i; \mathbf{s})\right]^2, \tag{3.7}$$

where $\{\mathbf{x}_i\}_{i=1}^{P}$ denotes the template's pixel set over which the likelihood of the image is evaluated. Here, $\mathscr{W}$ is a warping function, which acts as an interpolator for pixel locations between those defined by the shape $\mathbf{s}$. In other words, the shape defines a parameterisation that controls the spatial deformation field. This least squares data term is equivalent to assuming the distribution of appearance differences between the template and image follows that of an isotropic Gaussian[2]. This assumption is often invalid, especially for complex visual objects such as the human face, which generally exhibit large amounts of inter-subject appearance variability due to intrinsic variations or different imaging conditions. As such, a number of approaches to pairwise registration use more sophisticated measures of image-to-template similarity. For example, recent methods in variational optical flow use a robust error function to account for discontinuities in the appearance between images [22]. Image processing, such as evaluating error over the image gradients [23], is also utilised in these works, to minimise extrinsic lighting effects. Other approaches use statistical measures of similarity such as Mutual Information [93] and the correlation ratio [100] to handle differences in imaging modalities between the template and image.

Rather than choosing a more suitable similarity measure to account for the appearance variations between the template and image, another approach is to allow the template's appearance to deform, along with its shape. If the model for appearance deformation matches that of the visual object, then, at the optimal shape and appearance deformations, the template-to-image appearance residuals can be fully described by the measurement (image) noise. Assuming a Gaussian distribution on image noise, the likelihood function can be written as

$$p\left(\mathscr{I} \mid \mathbf{s}, \mathbf{s}^0, \mathscr{I}^0, \boldsymbol{\theta}\right) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2\sigma^2} \sum_{i=1}^{P} \left[\mathscr{A}(\mathbf{x}_i; \mathbf{w}) - \mathscr{I} \circ \mathscr{W}(\mathbf{x}_i; \mathbf{u}, \mathbf{v})\right]^2\right\}, \tag{3.8}$$

[2] An isotropic or spherical Gaussian distribution exhibits equal variances in all directions.

where σ^2 is the variance of image noise and:

$$\mathbf{s}^{(3n)} = \begin{bmatrix} \mathbf{s}^0 \\ \mathbf{0}^{(n)} \end{bmatrix} + \begin{bmatrix} \mathbf{u}^{(n)} \\ \mathbf{v}^{(n)} \\ \mathbf{w}^{(n)} \end{bmatrix} \tag{3.9}$$

is a redefinition of the shape into a set of deformations, defined with respect to the template, both in the spatial and appearance domains, defined by $(\mathbf{u}, \mathbf{v})$ and $\mathbf{w}$, respectively. Here,

$$\mathscr{A}(\mathbf{x}; \mathbf{w}) \colon \Re^2 \times \Re^n \to \Re = \mathscr{I}^0(\mathbf{x}) + \mathscr{M}(\mathbf{x}; \mathbf{w}) \tag{3.10}$$

is an appearance generating function, with:

$$\mathscr{M}(\mathbf{x}; \mathbf{w}) \colon \Re^2 \times \Re^n \to \Re \tag{3.11}$$

denoting the appearance deformation function, and:

$$\mathscr{W}(\mathbf{x}; \mathbf{u}, \mathbf{v}) \colon \Re^2 \times \Re^{2n} \to \Re^2 \tag{3.12}$$

is a warping function with $\mathscr{W}(\mathbf{x_i}; \mathbf{0}, \mathbf{0}) = \mathbf{x}_i$. Note that the appearance deformation function $\mathscr{M}$ is parameterised by n variables, the number of landmarks. Although the choice here may seem somewhat arbitrary, the reason for this choice will become clear in the discussion on deformation priors in the next section. With the parameterisation of the likelihood in Equation (3.8), maximising the MAP objective now involves a maximisation over the *spatial deformation* and *appearance deformation* simultaneously.

In general, however, the appropriate choice for the appearance generating function $\mathscr{A}$ may be difficult to deduce from domain knowledge. As such, in this study, the appearance generating function is used in conjunction with a robust similarity measure to define the image likelihood:

$$p\left(\mathscr{I} \mid \mathbf{s}, \mathbf{s}^0, \mathscr{I}^0, \boldsymbol{\theta}\right) = \frac{1}{\mathscr{Z}_L(\tau)} \exp\left\{-\tau \mathscr{D}_{PW}\left(\mathscr{I}; \mathbf{u}, \mathbf{v}, \mathbf{w}\right)\right\}, \tag{3.13}$$

where τ is the component of $\boldsymbol{\theta}$, which parameterises the image likelihood density. The partition (normalising) function $\mathscr{Z}_L$ ensures the form in Equation (3.13) is a PDF:

$$\mathscr{Z}_L(\tau) = \int_{\Re^n} \int_{\Re^n} \int_{\Re^n} \exp\left\{-\tau \mathscr{D}_{PW}\left(\mathscr{I}; \mathbf{u}, \mathbf{v}, \mathbf{w}\right)\right\} d\mathbf{u}\, d\mathbf{v}\, d\mathbf{w}. \tag{3.14}$$

The term in the exponential in Equation (3.13) is given by:

$$\mathscr{D}_{PW}(\mathscr{I}; \mathbf{u}, \mathbf{v}, \mathbf{w}) = \sum_{i=1}^{P} \psi\left(\left[\mathscr{A}(\mathbf{x}_i; \mathbf{w}) - \mathscr{I} \circ \mathscr{W}(\mathbf{x}_i; \mathbf{u}, \mathbf{v})\right]^2\right), \tag{3.15}$$

with ψ denoting the robust similarity measure. The idea here, is to choose the appearance generating function that can account for appearance differences which exhibit slow spatial changes, leaving the robust penaliser to account for local peaks.

3.3.2 Defining the Prior

With the reparameterisation of the correspondences as a set of spatial and appearance deformations, as stated in Equation (3.9), the prior in Equation (3.6) can now be written as:

$$p\left(\mathbf{s} \mid \mathbf{s}^0, \boldsymbol{\theta}\right) = p(\mathbf{u}|\alpha)\, p(\mathbf{v}|\beta)\, p(\mathbf{w}|\gamma) \tag{3.16}$$

where α, β and γ, also components of $\boldsymbol{\theta}$ (i.e. $\boldsymbol{\theta} = \{\tau, \alpha, \beta, \gamma\}$), define the parameterisations of the spatial and appearance prior densities. Following the convention set out in [66; 151], the components of $\boldsymbol{\theta}$ will be referred to in this dissertation as *hyperparameters*. In Equation (3.16), it is assumed that the x, y and appearance deformations are independent of each other. Also, it should be noted that the dependency of the prior on the template's shape is subsumed by the reparameterisation that now defines the problems in terms of deformations rather than shapes directly.

The choice of the prior PDFs should reflect domain knowledge about the types of deformations expected of the visual object. In the absence of any other information, a common choice is to assume that they are either smooth or piecewise smooth. This assumption, which has been used widely in the variational optical flow and stereo matching problems (see [61; 110; 145], for example), is based on the intuition that points that are close to each other move in similar ways. For visual objects, it also relates to topological rigidity since, if close points exhibit similar motion, then local topology will be *loosely* preserved.

Although the smoothness constraint has become the regulariser of choice for spatial deformations, the applicability of this assumption for appearance deformations of visual objects is yet to be verified. Numerical experiments motivating this assumption are presented in Section 3.6, hence it suffices to present here the intuitive reasons for choosing such a constraint for appearance deformations. There are two sources of appearance variation in visual objects: extrinsic and intrinsic. Extrinsic variations are the result of imaging conditions, such as lighting intensity and direction. Consider the case where the visual object is a projection of a 3D object onto the image plane. If the surface of the object is (piecewise) smooth, the appearance difference between the object's projection under ambient and directional light varies in a (piecewise) smooth fashion over the image (see Figure 3.2). Therefore, assuming (piecewise) smooth appearance deformations is equivalent to assuming the object's surface is (piecewise) smooth, which is a reasonable assumption in many cases, in particular for the human face, with which this thesis is mainly concerned. Intrinsic appearance variations, on the other hand, are more difficult to quantify. In general, they present abrupt changes in appearance, for example wrinkles on a human face. These variations can also be represented by a piecewise smooth function. However, the number of pieces will generally be quite large. For example, the variation of appearance *along* a wrinkle can be approximated by a smooth function.

To realise the smoothness constraint in a Bayesian framework, the priors $p(\mathbf{u}|\alpha)$, $p(\mathbf{v}|\beta)$ and $p(\mathbf{w}|\gamma)$ are characterised by Gibbs priors of the form:

$$p(\mathbf{p}|\nu) = \frac{1}{\mathscr{Z}_P(\nu)} \exp\{-\nu \mathscr{R}_{PW}(\mathbf{p})\}, \tag{3.17}$$

replacing $\mathbf{p}$ with $\mathbf{u}$,$\mathbf{v}$ or $\mathbf{w}$ and ν with α, β or γ for the x, y and appearance deformations,

Figure 3.2: Examples of piecewise smooth variations in appearance due to changes in extrinsic lighting conditions. Images are taken from the Yale Face Database B [51].

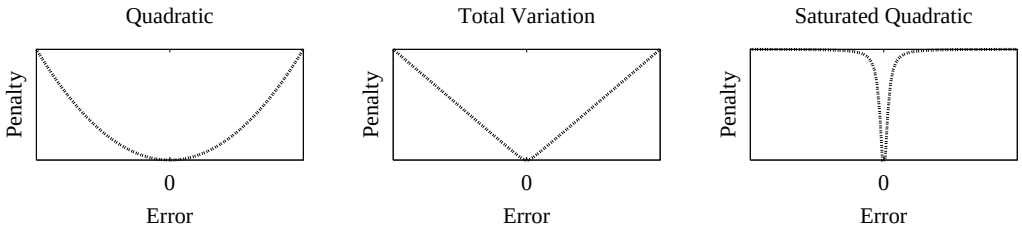


Figure 3.3: Examples of prior penalisers ϱ.

respectively. Here, $\mathscr{Z}_P$ is the prior's partition function, which depends on the parameter ν. The Gibbs energy $\mathscr{R}_{PW}$ is designed to impose (piecewise) smoothness on the deformations, taking the form:

$$\mathscr{R}_{PW}(\mathbf{p}) = \sum_{(i,j)\in\mathcal{E}} \kappa_{ij}\ \varrho\left(\left[\mathbf{p}_{(i)} - \mathbf{p}_{(j)}\right]^2\right), \tag{3.18}$$

where $\mathcal{E}$ denotes the edges of a 2D graph whose nodes are defined by the template's spatial shape $\mathbf{s}^0$. For example, in an irregular grid, the edges denote the sides of the triangles in the shape's triangulation. The constants κ_{ij} determine the contribution of an edge to the total energy through a monotonically decreasing kernel $\kappa_{ij} = \mathscr{K}(\mathbf{x}_i^0, \mathbf{x}_j^0)$. A common kernel to use here is the inverse of the squared distance between nodes [61]:

$$\mathscr{K}\left(\mathbf{x}_i^0, \mathbf{x}_j^0\right) : \Re^2 \times \Re^2 \to \Re = \frac{1}{\|\mathbf{x}_i^0 - \mathbf{x}_j^0\|^2}, \tag{3.19}$$

where $\mathbf{x}_i^0$ denotes the i^{th} landmark in the template's shape $\mathbf{s}^0$. The same form is used for all three axes of deformations. It should be noted here, that a separate template shape $\mathbf{s}^0$ can be used in the computation of $\mathcal{E}$ and subsequently, κ_{ij}, for the appearance deformation prior. This may involve a different number of template landmarks as well as their locations, which can be chosen to better represent appearance deformations of the visual object of interest. This option is not pursued in this study, however, where the same template shape for both the spatial and appearance deformation priors is maintained.

In Equation (3.18), ϱ denotes a distance function that penalises the squared difference

between the deformations of adjacent nodes. A choice of the identity function for ϱ allows only smooth deformations since the energy term, then, penalises discontinuities quadratically. For piecewise smooth deformations a number of different distance functions have been proposed. Two of these are illustrated in Figure 3.3, together with the identity function. Semi convex functions, such as the regularised variant of the L_1 norm:

$$\varrho\left(v^2;\epsilon\right) = \sqrt{v^2+\epsilon}, \tag{3.20}$$

which corresponds to the total variation regularisation [106], have been used widely in variational optical flow [22; 23; 110]. They allow a limited amount of abrupt changes in the deformation field, by virtue of their fixed rate of penalty, whilst being somewhat insensitive to the small regularisation parameter ϵ, often included for numerical reasons only. Distance functions that decrease the rate of penalty beyond a threshold, such as the saturated quadratic:

$$\varrho(v^2;\epsilon) = \frac{v^2}{\epsilon^2+v^2}, \tag{3.21}$$

positively favour the presence of edges in the deformation fields. As such, non-convex distance functions enforce the smoothness constraints less than their semi-convex counterpart. The resulting energy is also much more dependent on the threshold used, requiring its tuning in many applications. Although these two classes of robust distance functions are both capable of preserving discontinuities, semi-convex functions place more restraint on them. In the case of LDM correspondence learning, the number of discontinuities is relatively small compared to that in general optical flow or stereo matching for example, where there may be many independently moving objects in the scene as well as occlusional effects. As such, semi-convex functions may better approximate the deformations exhibited by the visual object in the case of LDM correspondence learning. Furthermore, due to the decreasing penalty rate of non-convex distance functions, optimisation here may be slower, since the magnitude of the errors is not directly reflected by the gradients. Finally, the use of a non-convex penaliser adds extra nonlinearities to an already nonlinear problem, increasing the likelihood of an optimisation terminating in a local minimum.

3.3.3 Parameterising Deformations

Despite utilising a pseudo-dense correspondence set, the appearance of an image that is used to evaluate the likelihood of an image in the MAP framework must, in general, be dense. A pseudo-dense representation of appearance will require a set of pixels within the template to be chosen *a-priori*, in order to assess the quality of the learnt correspondences. Due to intrinsic variations of appearance within a visual object, choosing salient pixels from the reference frame will often result in a sub-optimal formulation, since their homologous locations in other images may not be salient. Therefore, pseudo-dense LDMs require a warping function to interpolate the projections of the landmarks for all pixels within the template's valid domain. In effect, the warping function extends the deformation of LDM landmarks to a deformation over the whole appearance domain.

The optimal type of warping function to use here will generally depend on the visual object

of interest. Two of the most common examples are the thin plate spline [10] and the piecewise affine warp [83]. In this thesis, the piecewise affine warp will be considered exclusively, as it affords an efficient evaluation of the warp as well as of its gradient. For clarification, consider the linear warping function:

$$\mathscr{W}(\mathbf{x};\mathbf{s}) : \Re^2 \times \Re^{2n} \rightarrow \Re^2 = \mathbf{A}_{\mathbf{x}}\mathbf{s} \tag{3.22}$$

where $\mathbf{A}_{\mathbf{x}}^{(2\times 2n)}$ is an interpolation matrix, which can be precomputed for every valid location in the set:

$$\{\mathbf{x}_i\}_{i=1}^{P} \in \mathbf{\Omega} = \text{hull}\left\{\mathbf{s}^0\right\}. \tag{3.23}$$

Both the piecewise affine warp and the thin plate spline share this functional form[3]. However, in the case of the thin plate spline, the matrix $\mathbf{A}_{\mathbf{x}}$ is dense, whereas, for the piecewise affine warp it is extremely sparse (see Appendix A). In fact, in each row of the piecewise affine warp's $\mathbf{A}_{\mathbf{x}}$, only three entries are non-zero, each pertaining to one of the vertices of the triangle containing the location of interest in the template. Referring to Appendix A, and parameterising the warp using landmark deformations $(\mathbf{u},\mathbf{v})$, the explicit form of the piecewise affine warp is given by:

$$\mathscr{W}(\mathbf{x}_p;\mathbf{u},\mathbf{v}) = \mathbf{x}_p + \begin{bmatrix} 1-\alpha_p-\beta_p & \alpha_p & \beta_p & 0 & 0 & 0 \\ 0 & 0 & 0 & 1-\alpha_p-\beta_p & \alpha_p & \beta_p \end{bmatrix} \begin{bmatrix} \mathbf{u}_{(i)} \\ \mathbf{u}_{(j)} \\ \mathbf{u}_{(k)} \\ \mathbf{v}_{(i)} \\ \mathbf{v}_{(j)} \\ \mathbf{v}_{(k)} \end{bmatrix}, \tag{3.24}$$

where:

$$\mathbf{x}_p \in \text{tri}\left\{\mathbf{s}^0_{(2i-1:2i)}, \mathbf{s}^0_{(2j-1:2j)}, \mathbf{s}^0_{(2k-1:2k)}\right\}. \tag{3.25}$$

The values for α_p and β_p are as given in Equation (A.2), specialised to the p^{th} valid location in the template. From this, it is clear that the computational savings of the piecewise affine warp compared to the thin plate spline grows linearly with the number of landmarks in the model. Finally, if the correspondences are later used to build an AAM, the use of the piecewise affine warp in learning will better couple the correspondences with their utility, later, for AAM fitting.

To account for differences between the template and the image, the pairwise method proposed in this chapter allows the template's appearance to deform along with its shape. The choice of how the appearance generating function is parameterised should reflect the type of appearance differences expected within the training set. However, it should also be noted that the deformable template is used in conjunction with a robust penaliser over the differences between the synthesised appearance and that of the object in the image. As such, the system has the capacity to tolerate localised extreme errors. Therefore, a parameterisation that accounts for appearance differences which vary *slowly* through the spatial domain, should be chosen here.

[3] Strictly speaking, the thin plate spline also has an affine term, but it does not effect the exposition in this section.

For this, a piecewise affine appearance deformation model seems a natural choice. Following the reformulation of the piecewise affine warp as a scalar valued function in Appendix A, the piecewise affine appearance generating function takes the form:

$$\mathscr{M}(\mathbf{x}_p;\mathbf{w}) = \begin{bmatrix}1-\alpha_p-\beta_p & \alpha_p & \beta_p\end{bmatrix}\begin{bmatrix}\mathbf{w}_{(i)}\\ \mathbf{w}_{(j)}\\ \mathbf{w}_{(k)}\end{bmatrix} \tag{3.26}$$

for $\mathbf{x}_p$ as in Equation (3.25), where $\mathbf{w}_{(i)}$ denotes the appearance deformation at location $\mathbf{s}^0_{(2i-1:2i)}$ in the template (similarly for $\mathbf{w}_{(j)}$ and $\mathbf{w}_{(k)}$).

3.4 Marginalised Maximum Likelihood Estimation

The MML estimation of the densities' hyperparameters involves maximising the data likelihood, with marginalisation over the shape, given in Equation (3.5). Using the reparameterisation in Equation (3.9), this can be written:

$$p\left(\mathscr{I},\mathscr{I}^0,\mathbf{s}^0 \mid \boldsymbol{\theta}\right) \propto \int_{\Re^n}\int_{\Re^n}\int_{\Re^n} p(\mathscr{I}|\mathbf{u},\mathbf{v},\mathbf{w},\tau)\,p(\mathbf{u}|\alpha)\,p(\mathbf{v}|\beta)\,p(\mathbf{w}|\gamma)\,d\mathbf{u}\,d\mathbf{v}\,d\mathbf{w}. \tag{3.27}$$

Examining the forms of the image likelihood and deformation priors in Equations (3.13) and (3.17), it is clear that an analytic form of this integral does not exist in general. This is the case, even if ψ in Equation (3.15) and ϱ in Equation (3.18) are both set to the identity function[4], due to the generally nonlinear relationship between the image intensities and the deformations in Equation (3.15). Therefore, in order to obtain an estimate of the optimal parameterisations of the densities without resorting to numerical integration methods, the densities in Equation (3.27) must be approximated in such a way that the integral affords an analytic solution. For this, the likelihood and prior densities are approximated with Gaussians. Since the improper integral of a Gaussian can be evaluated analytically, the integral of the joint density, which becomes the product of Gaussians, can then be found.

3.4.1 Gaussian Approximated Prior

Consider first the prior terms given by the form in Equation (3.17). When assuming strictly smooth deformations, the prior densities are already in Gaussian form:

$$p(\mathbf{p}|\nu) = \frac{1}{\mathscr{Z}_P(\nu)}\exp\left\{-\frac{1}{2}\mathbf{p}^T(2\nu\mathbf{H})\mathbf{p}\right\}, \tag{3.28}$$

again replacing $\mathbf{p}$ with $\mathbf{u}$,$\mathbf{v}$ or $\mathbf{w}$, and ν with α, β or γ for the x, y and appearance deformations, respectively. The Gaussian's covariance is $(2\mathbf{H})^{-1}$, with:

$$\mathbf{H} = \sum_{(i,j)\in\mathcal{E}} \kappa_{ij}\left(\mathbf{1}_i^{(n)}-\mathbf{1}_j^{(n)}\right)\left(\mathbf{1}_i^{(n)}-\mathbf{1}_j^{(n)}\right)^T, \tag{3.29}$$

[4]This scenario corresponds to assuming strictly smooth deformations with Gaussian image noise.

where $\mathbf{1}_i^{(n)}$ is a n-length vector with 1 in the i^{th} entry and zero everywhere else. The partition function is given by:

$$\mathscr{Z}(\nu) = \int_{\Re^n} \exp\left\{-\frac{1}{2}\mathbf{p}^T(2\nu\mathbf{H})\mathbf{p}\right\} d\mathbf{p} = \frac{(2\pi)^{\frac{n}{2}}}{\det(2\nu\mathbf{H})^{\frac{1}{2}}} \propto \nu^{-\frac{n}{2}}. \tag{3.30}$$

However, when describing piecewise smooth deformations, through the utilisation of a robust penalty function in the energy term, the prior does not take on a Gaussian form. In this case, the following change of variable is used:

$$\mathbf{p} = \mathbf{p}^c + \Delta\mathbf{p} \quad \text{and} \quad d\mathbf{p} = d\Delta\mathbf{p}, \tag{3.31}$$

where $\mathbf{p}^c$ denotes the current estimate of the deformation and $\Delta\mathbf{p}$ denotes some perturbation from the current estimates. In nonrigid registration involving a nonlinear distance function, a typical approach is to linearise the distance function using the current deformation estimates $\mathbf{p}^c$, then solve the linearised form with respect to the perturbations $\Delta\mathbf{p}$ [4]. The same idea can be applied here, whereby taking a first order Taylor expansion of ϱ about the current squared deformation residuals, results in:

$$\varrho\left(\left[\mathbf{p}_{(i)}^c + \Delta\mathbf{p}_{(i)} - \mathbf{p}_{(j)}^c - \Delta\mathbf{p}_{(j)}\right]^2\right) \approx \varrho\left(r_{ij}^2\right) + \nabla\varrho\left(r_{ij}^2\right)\left[\Delta\mathbf{p}^T\mathbf{H}_{ij}\Delta\mathbf{p} + 2\mathbf{h}_{ij}^T\Delta\mathbf{p}\right], \tag{3.32}$$

where $\nabla\varrho\left(r_{ij}^2\right)$ is the derivative of the robust penaliser, evaluated at r_{ij}^2, and:

$$r_{ij} = \mathbf{p}_{(i)}^c - \mathbf{p}_{(j)}^c \quad , \quad \mathbf{h}_{ij} = r_{ij}\left(\mathbf{1}_i^{(n)} - \mathbf{1}_j^{(n)}\right) \quad \text{and} \quad \mathbf{H}_{ij} = \left(\mathbf{1}_i^{(n)} - \mathbf{1}_j^{(n)}\right)\left(\mathbf{1}_i^{(n)} - \mathbf{1}_j^{(n)}\right)^T. \tag{3.33}$$

Substituting this into Equation (3.18) and completing the square, the regularisation energy can be written:

$$\mathscr{R}_{PW}(\mathbf{p}) \approx (\Delta\mathbf{p} - \bar{\mathbf{p}})^T\,\mathbf{H}\,(\Delta\mathbf{p} - \bar{\mathbf{p}}) + C_P, \tag{3.34}$$

where:

$$\bar{\mathbf{p}} = -\mathbf{H}^{-1}\mathbf{h} \quad \text{and} \quad C_P = r - \mathbf{h}^T\mathbf{H}^{-1}\mathbf{h}. \tag{3.35}$$

Here, the following collected terms have been used:

$$r = \sum_{(i,j)\in\mathcal{E}} \kappa_{ij}\varrho\left(r_{ij}^2\right) \quad , \quad \mathbf{h} = \sum_{(i,j)\in\mathcal{E}} \kappa_{ij}\nabla\varrho\left(r_{ij}^2\right)\mathbf{h}_{ij} \quad \text{and} \quad \mathbf{H} = \sum_{(i,j)\in\mathcal{E}} \kappa_{ij}\nabla\varrho\left(r_{ij}^2\right)\mathbf{H}_{ij}. \tag{3.36}$$

With this linearisation, the Gaussian approximation of the prior density takes the form:

$$p(\mathbf{p}|\nu) \approx \frac{1}{\tilde{\mathscr{Z}}_P(\nu)} \exp\{-\nu C_P\} \exp\left\{-\frac{1}{2}(\Delta\mathbf{p} - \bar{\mathbf{p}})^T(2\nu\mathbf{H})(\Delta\mathbf{p} - \bar{\mathbf{p}})\right\}, \tag{3.37}$$

where the partition function is now given by:

$$\tilde{\mathscr{Z}}_P(\nu) = \exp\{-\nu C_P\} \int_{\Re^n} \exp\left\{-\frac{1}{2}\left(\Delta\mathbf{p}-\bar{\mathbf{p}}\right)^T (2\nu\mathbf{H})\left(\Delta\mathbf{p}-\bar{\mathbf{p}}\right)\right\} d\Delta\mathbf{p} \tag{3.38}$$

$$= \exp\{-\nu C_P\} \frac{(2\pi)^{\frac{n}{2}}}{\det(2\nu\mathbf{H})^{\frac{1}{2}}} \propto \nu^{-\frac{n}{2}} \exp\{-\nu C_P\}, \tag{3.39}$$

which finally leads to:

$$p(\mathbf{p}|\nu) \propto \nu^{\frac{n}{2}} \exp\left\{-\frac{1}{2}\left(\Delta\mathbf{p}-\bar{\mathbf{p}}\right)^T (2\nu\mathbf{H})\left(\Delta\mathbf{p}-\bar{\mathbf{p}}\right)\right\}. \tag{3.40}$$

This is a multivariate Gaussian distribution over the deformations $\Delta\mathbf{p}$.

3.4.2 Gaussian Approximated Likelihood

Let us now consider the likelihood term in Equation (3.27). As previously discussed, unlike the prior, the image likelihood is not in a Gaussian form with respect to the deformations, regardless of how ψ is defined. For this, the same change of variable is used as in the treatment of the prior. Taking a first order Taylor expansion of the warped image and the appearance generator about the current parameters results in:

$$\mathscr{I} \circ \mathscr{W}(\mathbf{x}_i; \mathbf{u}^c + \Delta\mathbf{u}, \mathbf{v}^c + \Delta\mathbf{v}) \approx \mathscr{I} \circ \mathscr{W}(\mathbf{x}_i; \mathbf{u}^c, \mathbf{v}^c) + \mathbf{J}_i\,[\Delta\mathbf{u}; \Delta\mathbf{v}] \tag{3.41}$$

$$\mathscr{A}(\mathbf{x}_i; \mathbf{w}^c + \Delta\mathbf{w}) \approx \mathscr{A}(\mathbf{x}_i; \mathbf{w}^c) + \mathbf{A}_i\,\Delta\mathbf{w}, \tag{3.42}$$

where:

$$\mathbf{A}_i = \frac{\partial \mathscr{M}\left(\mathbf{x}_i; \mathbf{w}^c\right)}{\partial \mathbf{w}} \quad \text{and} \quad \mathbf{J}_i = \nabla_{\mathbf{x}}\mathscr{I}(\vec{\mathbf{x}}_i)\left[\frac{\partial \mathscr{W}(\mathbf{x}_i;\mathbf{u}^c,\mathbf{v}^c)}{\partial \mathbf{u}} \quad \frac{\partial \mathscr{W}(\mathbf{x}_i;\mathbf{u}^c,\mathbf{v}^c)}{\partial \mathbf{v}}\right]. \tag{3.43}$$

Here, $\vec{\mathbf{x}}_i = \mathscr{W}(\mathbf{x}_i; \mathbf{u}^c, \mathbf{v}^c)$ is the location in the image frame that corresponds to location $\mathbf{x}_i$ in the template and $\nabla_{\mathbf{x}}\mathscr{I}(\vec{\mathbf{x}}_i)$ is the image's spatial derivative at that location. Letting $\mathbf{q} = [\Delta\mathbf{u}; \Delta\mathbf{v}; \Delta\mathbf{w}]$, the likelihood's energy term in Equation (3.15) can be written:

$$\mathscr{D}_{PW}(\mathscr{I}; \mathbf{q}) \approx \sum_{i=1}^{P} \psi\left(\mathbf{q}^T\mathbf{d}_i^T\mathbf{d}_i\mathbf{q} + 2e_i\mathbf{d}_i\mathbf{q} + e_i^2\right), \tag{3.44}$$

where

$$e_i = \mathscr{A}(\mathbf{x}_i; \mathbf{w}^c) - \mathscr{I} \circ \mathscr{W}(\mathbf{x}_i; \mathbf{u}^c, \mathbf{v}^c) \quad \text{and} \quad \mathbf{d}_i = [\,-\mathbf{J}_i \quad \mathbf{A}_i\,]. \tag{3.45}$$

Taking another first order Taylor expansion, now of the robust function ψ, around the current appearance residuals, and completing the square, Equation (3.15) can be approximated by:

$$\mathscr{D}_{PW}(\mathscr{I}; \mathbf{q}) \approx \left(\mathbf{q}-\bar{\mathbf{q}}\right)^T \mathbf{D}\left(\mathbf{q}-\bar{\mathbf{q}}\right) + C_L, \tag{3.46}$$

where:

$$\mathbf{D} = \sum_{i=1}^{P} \nabla\psi\left(e_i^2\right) \mathbf{d}_i^T \mathbf{d}_i \tag{3.47}$$

$$\bar{\mathbf{q}} = -\mathbf{D}^{-1} \left(\sum_{i=1}^{P} \nabla\psi\left(e_i^2\right)\ e_i\, \mathbf{d}_i \right) \tag{3.48}$$

$$C_L = \left[\sum_{i=1}^{P} \psi\left(e_i^2\right) \right] - \left[\sum_{i=1}^{P} \nabla\psi\left(e_i^2\right)\ e_i\, \mathbf{d}_i \right]^T \mathbf{D}^{-1} \left[\sum_{i=1}^{P} \nabla\psi\left(e_i^2\right)\ e_i\, \mathbf{d}_i \right]. \tag{3.49}$$

The Gaussian approximation of the image likelihood, then, takes the form:

$$p(\mathscr{I} \mid \mathbf{q}\,, \tau) \approx \frac{1}{\tilde{\mathscr{Z}}_L(\tau)} \exp\{-\tau C_L\} \exp\left\{ -\frac{1}{2} \left(\mathbf{q} - \bar{\mathbf{q}}\right)^T (2\tau\mathbf{D}) \left(\mathbf{q} - \bar{\mathbf{q}}\right) \right\}, \tag{3.50}$$

where the partition function is now given by:

$$\mathscr{Z}_L(\tau) = \exp\left\{-\tau C_L\right\} \int_{\Re^{3n}} \exp\left\{ -\frac{1}{2} \left(\mathbf{q} - \bar{\mathbf{q}}\right)^T (2\tau\mathbf{D}) \left(\mathbf{q} - \bar{\mathbf{q}}\right) \right\} d\mathbf{q} \tag{3.51}$$

$$= \exp\left\{-\tau C_L\right\} \frac{(2\pi)^{\frac{3n}{2}}}{\det(2\tau\mathbf{D})^{\frac{1}{2}}} \propto \tau^{-\frac{3n}{2}} \exp\left\{-\tau C_L\right\}, \tag{3.52}$$

which finally leads to:

$$p(\mathscr{I} \mid \mathbf{q}\,, \tau) \propto \tau^{\frac{3n}{2}} \exp\left\{ -\frac{1}{2} \left(\mathbf{q} - \bar{\mathbf{q}}\right)^T (2\tau\mathbf{D}) \left(\mathbf{q} - \bar{\mathbf{q}}\right) \right\}. \tag{3.53}$$

As with the prior density, the linearisation of the appearance generator, and subsequently the robust penaliser, results in a Gaussian density over the deformations $\mathbf{q}$.

3.4.3 Estimation through Expectation Maximisation

With the Gaussian approximations for the image likelihood and deformation priors in Equations (3.53) and (3.40), respectively, Equation (3.27) can be written:

$$p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0 \mid \boldsymbol{\theta}\right) \propto \left(\alpha\beta\gamma\tau^3\right)^{\frac{n}{2}} \exp\left\{-C\right\} \int_{\Re^{3n}} \exp\left\{ -\left(\mathbf{q} - \boldsymbol{\mu}\right)^T \boldsymbol{\Sigma} \left(\mathbf{q} - \boldsymbol{\mu}\right) \right\} d\mathbf{q}, \tag{3.54}$$

where:

$$\boldsymbol{\Sigma} = \tau\mathbf{D} + \begin{bmatrix} \alpha\mathbf{H}_u & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \beta\mathbf{H}_v & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \gamma\mathbf{H}_w \end{bmatrix} \quad , \quad \boldsymbol{\mu} = \boldsymbol{\Sigma}^{-1} \left(\tau\mathbf{D}\bar{\mathbf{q}} + \begin{bmatrix} \alpha\mathbf{H}_u\bar{\mathbf{u}} \\ \beta\mathbf{H}_v\bar{\mathbf{v}} \\ \gamma\mathbf{H}_w\bar{\mathbf{w}} \end{bmatrix} \right) \tag{3.55}$$

$$\text{and} \quad C = \alpha\bar{\mathbf{u}}^T\mathbf{H}_u\bar{\mathbf{u}} + \beta\bar{\mathbf{v}}^T\mathbf{H}_v\bar{\mathbf{v}} + \gamma\bar{\mathbf{w}}^T\mathbf{H}_w\bar{\mathbf{w}} + \tau\bar{\mathbf{q}}^T\mathbf{D}\bar{\mathbf{q}} - \boldsymbol{\mu}^T\boldsymbol{\Sigma}\boldsymbol{\mu}. \tag{3.56}$$

Here, $\bar{\mathbf{u}}$ and $\mathbf{H}_u$ take the forms in Equation (3.35) and Equation (3.36), respectively, specialised to the case of x-spatial deformation, and similarly for the case of y-spatial and appearance deformations. Evaluating the integral in Equation (3.54) results in:

$$p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0 \mid \boldsymbol{\theta}\right) \propto \left(\alpha\beta\gamma\tau^3\right)^{\frac{n}{2}} \exp\{-C\}\det(\boldsymbol{\Sigma})^{-\frac{1}{2}}. \tag{3.57}$$

Finally, taking its negative logarithm, the cost function to be minimised, with respect to the parameterisation $\boldsymbol{\theta}$, is given by:

$$\mathscr{E}_{MML}(\tau, \alpha, \beta, \gamma) = C - \frac{n}{2}\ln\{\alpha\} - \frac{n}{2}\ln\{\beta\} - \frac{n}{2}\ln\{\gamma\} - \frac{3n}{2}\ln\{\tau\} + \frac{1}{2}\ln\{\det(\boldsymbol{\Sigma})\}. \tag{3.58}$$

Although the form of this cost function is quite complex, since it is an optimisation over only four parameters, a non-gradient based optimiser, such as the simplex [94], can be utilised to obtain a solution.

An alternative to utilising a general purpose optimiser for minimising Equation (3.58) is to utilise the Expectation Maximisation (EM) algorithm [14]. Treating the deformations $\mathbf{q}$ as hidden variables and the hyperparameters $\boldsymbol{\theta} = [\tau; \alpha; \beta; \gamma]$ as parameters, the EM algorithm first finds the expected data log-likelihood:

$$\begin{aligned}\mathscr{Q}(\tilde{\boldsymbol{\theta}}; \boldsymbol{\theta}) &= E_{p\left(\mathbf{q} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \tilde{\boldsymbol{\theta}}\right)}\left[\ \ln\left\{p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{q} \mid \boldsymbol{\theta}\right)\right\} \ \mid \tilde{\boldsymbol{\theta}}\ \right] \\ &= \int_{\Re^{3n}} p\left(\mathbf{q} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \tilde{\boldsymbol{\theta}}\right) \ln\left\{p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{q} \mid \boldsymbol{\theta}\right)\right\} d\mathbf{s},\end{aligned} \tag{3.59}$$

given the current estimate of the hyperparameters $\tilde{\boldsymbol{\theta}}$. Through the utility of Jensen's inequality, this objective function can be shown to upper bound the log of the likelihood in Equation (3.57), and touches it at the current estimate of the hyperparameters $\tilde{\boldsymbol{\theta}}$. As such, alternating an E-step (expectation), which defines $p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{q} \mid \boldsymbol{\theta}\right)$ using the current hyperparameter estimates, with an M-step (maximisation) over Equation (3.59), the algorithm is guaranteed to converge to a local optimum.

Using the identities:

$$p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{q} \mid \boldsymbol{\theta}\right) = p\left(\mathscr{I} \mid \mathbf{q}, \mathscr{I}^0, \mathbf{s}^0, \boldsymbol{\theta}\right) p\left(\mathbf{q} \mid \mathbf{s}^0, \boldsymbol{\theta}\right) \tag{3.60}$$

and

$$p\left(\mathbf{q} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \boldsymbol{\theta}\right) = \frac{p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{q} \mid \boldsymbol{\theta}\right)}{\int_{\Re^{3n}} p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{q} \mid \boldsymbol{\theta}\right) d\mathbf{q}}, \tag{3.61}$$

the objective of the M-step takes the form:

$$\mathscr{Q}(\tilde{\boldsymbol{\theta}}; \boldsymbol{\theta}) = \int_{\Re^{3n}} \frac{p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{q} \mid \boldsymbol{\theta}\right)}{\int_{\Re^{3n}} p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{q} \mid \boldsymbol{\theta}\right) d\mathbf{q}} \ln\left\{p\left(\mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \mathbf{q} \mid \boldsymbol{\theta}\right)\right\} d\mathbf{q}. \tag{3.62}$$

Using the Gaussian approximation for the image likelihood and deformation priors derived in Sections (3.4.2) and (3.4.1), respectively, and substituting into Equation (3.62), the posterior

over the deformation updates takes the form:

$$p\left(\mathbf{q} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \tilde{\boldsymbol{\theta}}\right) \propto \frac{\left(\alpha\beta\gamma\tau^3\right)^{\frac{n}{2}} \exp\left\{-(\mathbf{q}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}(\mathbf{q}-\boldsymbol{\mu}) - C\right\}}{\left(\alpha\beta\gamma\tau^3\right)^{\frac{n}{2}} \exp\{-C\} \det(\boldsymbol{\Sigma})^{-\frac{1}{2}}} = \mathcal{N}\left(\mathbf{q}; \boldsymbol{\mu}, (2\boldsymbol{\Sigma})^{-1}\right), \tag{3.63}$$

where $\mathcal{N}\left(\mathbf{q}; \boldsymbol{\mu}, (2\boldsymbol{\Sigma})^{-1}\right)$ denotes the Gaussian PDF over the deformation updates $\mathbf{q}$, with mean $\boldsymbol{\mu}$ and covariance $(2\boldsymbol{\Sigma})^{-1}$, both given in Equation (3.55). Using the same approximation for the data log-likelihood, the M-step involves maximising:

$$\begin{aligned}\mathscr{Q}(\tilde{\boldsymbol{\theta}}; \boldsymbol{\theta}) = \frac{3n}{2}\ln\{\tau\} - \tau\, E_{p\left(\mathbf{q}|\mathscr{I},\mathscr{I}^0,\mathbf{s}^0,\tilde{\boldsymbol{\theta}}\right)}\left[\,(\mathbf{q}-\bar{\mathbf{q}})^T\mathbf{D}(\mathbf{q}-\bar{\mathbf{q}})\,\right] + \\ \frac{n}{2}\ln\{\alpha\} - \alpha\, E_{p\left(\mathbf{q}|\mathscr{I},\mathscr{I}^0,\mathbf{s}^0,\tilde{\boldsymbol{\theta}}\right)}\left[\,(\Delta\mathbf{u}-\bar{\mathbf{u}})^T\mathbf{H}_u(\Delta\mathbf{u}-\bar{\mathbf{u}})\,\right] + \\ \frac{n}{2}\ln\{\beta\} - \beta\, E_{p\left(\mathbf{q}|\mathscr{I},\mathscr{I}^0,\mathbf{s}^0,\tilde{\boldsymbol{\theta}}\right)}\left[\,(\Delta\mathbf{v}-\bar{\mathbf{v}})^T\mathbf{H}_v(\Delta\mathbf{v}-\bar{\mathbf{v}})\,\right] + \\ \frac{n}{2}\ln\{\gamma\} - \gamma\, E_{p\left(\mathbf{q}|\mathscr{I},\mathscr{I}^0,\mathbf{s}^0,\tilde{\boldsymbol{\theta}}\right)}\left[\,(\Delta\mathbf{w}-\bar{\mathbf{w}})^T\mathbf{H}_w(\Delta\mathbf{w}-\bar{\mathbf{w}})\,\right].\end{aligned} \tag{3.64}$$

Taking the derivative of $\mathscr{Q}$ with respect to τ and equating to zero, the image likelihood rate that maximises the expected data log-likelihood is given by:

$$\tau = \frac{3n}{2\, E_{p\left(\mathbf{q}|\mathscr{I},\mathscr{I}^0,\mathbf{s}^0,\tilde{\boldsymbol{\theta}}\right)}\left[\,(\mathbf{q}-\bar{\mathbf{q}})^T\mathbf{D}(\mathbf{q}-\bar{\mathbf{q}})\,\right]} \quad . \tag{3.65}$$

Since $\mathcal{N}$ is a Gaussian, the relation [73]: $E_{\mathcal{N}}\left[\|\mathbf{e}-\boldsymbol{\Phi}\mathbf{q}\|^2\right] = \|\mathbf{e}-\boldsymbol{\Phi}\boldsymbol{\mu}\|^2 + \text{tr}\left\{\boldsymbol{\Phi}^T\boldsymbol{\Phi}\boldsymbol{\Sigma}\right\}$ can be utilised, to get:

$$\tau = \frac{3n}{2}\left[(\boldsymbol{\mu}-\bar{\mathbf{q}})^T\,\mathbf{D}\,(\boldsymbol{\mu}-\bar{\mathbf{q}}) + \frac{1}{2}\text{tr}\left\{\mathbf{D}\boldsymbol{\Sigma}^{-1}\right\}\right]^{-1}. \tag{3.66}$$

Similarly, the prior hyperparameters that maximise the expected data log-likelihood are given by:

$$\alpha = \frac{n}{2}\left[(\mathbf{P}_u\boldsymbol{\mu}-\bar{\mathbf{u}})^T\,\mathbf{H}_u\,(\mathbf{P}_u\boldsymbol{\mu}-\bar{\mathbf{u}}) + \frac{1}{2}\text{tr}\left\{\mathbf{H}_u\mathbf{P}_u\boldsymbol{\Sigma}^{-1}\mathbf{P}_u^T\right\}\right]^{-1} \tag{3.67}$$

$$\beta = \frac{n}{2}\left[(\mathbf{P}_v\boldsymbol{\mu}-\bar{\mathbf{v}})^T\,\mathbf{H}_v\,(\mathbf{P}_v\boldsymbol{\mu}-\bar{\mathbf{v}}) + \frac{1}{2}\text{tr}\left\{\mathbf{H}_v\mathbf{P}_v\boldsymbol{\Sigma}^{-1}\mathbf{P}_v^T\right\}\right]^{-1} \tag{3.68}$$

$$\gamma = \frac{n}{2}\left[(\mathbf{P}_w\boldsymbol{\mu}-\bar{\mathbf{w}})^T\,\mathbf{H}_w\,(\mathbf{P}_w\boldsymbol{\mu}-\bar{\mathbf{w}}) + \frac{1}{2}\text{tr}\left\{\mathbf{H}_w\mathbf{P}_w\boldsymbol{\Sigma}^{-1}\mathbf{P}_w^T\right\}\right]^{-1} \tag{3.69}$$

where:

$$\mathbf{P}_u = \begin{bmatrix}\mathbf{I}^{(n\times n)} & \mathbf{0}^{(n\times n)} & \mathbf{0}^{(n\times n)}\end{bmatrix} \tag{3.70}$$

$$\mathbf{P}_v = \begin{bmatrix}\mathbf{0}^{(n\times n)} & \mathbf{I}^{(n\times n)} & \mathbf{0}^{(n\times n)}\end{bmatrix} \tag{3.71}$$

$$\mathbf{P}_w = \begin{bmatrix}\mathbf{0}^{(n\times n)} & \mathbf{0}^{(n\times n)} & \mathbf{I}^{(n\times n)}\end{bmatrix}. \tag{3.72}$$

With this, a summary of the complete pairwise learning approach is outlined in Algorithm 2.

Algorithm 2 Pairwise Correspondence Learning

Require: $\{\mathscr{I}^0, \mathbf{s}^0\}$ (template), $\{\mathscr{I}_i\}_{i=1}^N$ (images), $\{\mathbf{u}_i, \mathbf{v}_i, \mathbf{w}_i\}_{i=1}^N$ (initial deformation estimates), $\{\tau_i, \alpha_i, \beta_i, \gamma_i\}_{i=1}^N$ (initial hyperparameter estimates), and N_{EM} (number of EM-algorithm steps)

1: **for** $i = 1$ to N **do**
2: **while** !converged $\{\mathbf{u}_i, \mathbf{v}_i, \mathbf{w}_i\}$ **do**
3: Minimise Equation (3.75) over $\{\mathbf{u}_i, \mathbf{v}_i, \mathbf{w}_i\}$ {see Algorithm 3 in Section 3.5}
4: Build Gaussian approximated likelihood and priors {see Sections 3.4.2 and 3.4.1}
5: **for** $j = 1$ to N_{EM} **do**
6: E-step: Compute $\{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$ {Equation (3.55)}
7: M-step: Compute $\{\tau, \alpha, \beta, \gamma\}$ {Equations (3.66), (3.67), (3.68) and (3.69)}
8: **end for**
9: **end while**
10: Set $\mathbf{s}_i = \mathbf{s}^0 + \mathbf{R}\,[\,\mathbf{u}_i\,;\,\mathbf{v}_i\,]$
11: **end for**
12: **return** $\{\mathbf{s}_i\}_{i=1}^N$

Note that in step 10 of the algorithm, $\mathbf{R}$ is chosen to alternate the x and y deformations in accordance with the format of the shape vector given in Equation (3.2).

As a final note on the MML method proposed here, the difference between its derivation of the objective for the hyperparmeters compared with a similar approach proposed in [66] will be highlighted. In that work, an integrable form for the component within the integral in Equation (3.27) is obtained by taking an *incomplete* second order Taylor expansion about the current parameter estimates[5]. The result is a much simplified objective at the expense of a poorer quality of approximation. Furthermore, interactions between connected points within the affinity matrix are ignored, simplifying further the objective to minimise. Finally, the optimisation of the marginalised likelihood in this work is performed directly, rather than through an EM procedure, since the assumptions made regarding the expansion as well as the affinity matrix result in the updates taking particularly simple forms.

3.5 Maximising the Pairwise Posterior

With the parameterisations of the image likelihood and deformation priors described in Sections 3.3.1 and 3.3.2, respectively, and the current estimate of the hyperparameters obtained from the EM procedure outlined in Section 3.4.3, the MAP objective for pairwise correspondence learning involves maximising:

$$p(\mathbf{s} \mid \mathscr{I}, \mathscr{I}^0, \mathbf{s}^0, \boldsymbol{\theta}) \propto p(\mathscr{I}|\mathbf{u}, \mathbf{v}, \mathbf{w}, \tau)\; p(\mathbf{u}|\alpha)\; p(\mathbf{v}|\beta)\; p(\mathbf{w}|\gamma) \tag{3.74}$$

[5] In [66], the approximation to the energy term within the exponential of the joint likelihood is taken as:

$$\mathscr{F}(\mathbf{p}) = \mathscr{F}(\mathbf{p}^c) + (\mathbf{p} - \mathbf{p}^c)\mathbf{Q}(\mathbf{p} - \mathbf{p}^c), \tag{3.73}$$

where $\mathscr{F}(\mathbf{p}) = \sigma\mathscr{D}(\mathbf{p}) + \gamma\mathscr{R}(\mathbf{p})$ and $\mathbf{Q}$ is the Hessian of the combined energy terms (note that no appearance deformation is used here). By not considering the first order term in the expansion, the energy term is already in the canonical quadratic from, hence a procedure to complete the square is not required.

with respect to the deformations. Taking its negative logarithm and substituting the forms in Equations (3.13) and (3.17), the pairwise MAP cost function to be minimised can be written:

$$\mathscr{E}_{MAP}(\mathbf{u},\mathbf{v},\mathbf{w}) = \mathscr{D}_{PW}(\mathscr{I};\mathbf{u},\mathbf{v},\mathbf{w}) + \lambda_x \mathscr{R}_{PW}(\mathbf{u}) + \lambda_y \mathscr{R}_{PW}(\mathbf{v}) + \lambda_a \mathscr{R}_{PW}(\mathbf{w}), \quad (3.75)$$

where $\mathscr{D}_{PW}$ and $\mathscr{R}_{PW}$ are defined as in Equations (3.15) and (3.18), respectively, and:

$$\lambda_x = \frac{\alpha}{\tau} \quad , \quad \lambda_y = \frac{\beta}{\tau} \quad \text{and} \quad \lambda_a = \frac{\gamma}{\tau}. \quad (3.76)$$

This cost function takes the same form as the typical regularised data fitting problem in Equation (3.1), where three regularisation parameters are now involved. Minimising this error function constitutes a nonlinear optimisation over the deformation parameters. Although second order methods, such as the Newton method, may be applicable here, it requires the computation of the Hessian matrix, which is computationally expensive. Instead, an approach similar to that proposed in [6] is utilised, which alternates between linearising the robust functions and solving a weighted nonlinear least squares problem, repeatedly until convergence.

Using the change of variable:

$$\mathbf{u} = \mathbf{u}^c + \Delta\mathbf{u} \quad , \quad \mathbf{v} = \mathbf{v}^c + \Delta\mathbf{v} \quad \text{and} \quad \mathbf{w} = \mathbf{w}^c + \Delta\mathbf{w}, \quad (3.77)$$

where $\{\mathbf{u}^c, \mathbf{v}^c, \mathbf{w}^c\}$ denotes the current estimates of the deformations and $\{\Delta\mathbf{u}, \Delta\mathbf{v}, \Delta\mathbf{w}\}$ denotes the desired updates, by linearising the cropped image and subsequently, all robust functions as in Sections 3.4.1 and 3.4.2, Equation (3.75) can be approximated by:

$$\mathscr{E}_{MAP}(\mathbf{u},\mathbf{v},\mathbf{w}) \approx \Delta\mathbf{q}^T\mathbf{H}_a\Delta\mathbf{q} + 2\mathbf{h}_a^T\Delta\mathbf{q} + \sum_{\mathbf{z}=\{\mathbf{u},\mathbf{v},\mathbf{w}\}} \left(\Delta\mathbf{z}^T\mathbf{H}_\mathbf{z}\Delta\mathbf{z} + 2\mathbf{h}_\mathbf{z}^T\Delta\mathbf{z}\right) + C, \quad (3.78)$$

where $\mathbf{q} = [\mathbf{u};\mathbf{v};\mathbf{w}]$. Here, C is a constant, which does not depend on the deformation updates, and:

$$\mathbf{h}_a = \sum_{i=1}^{P} e_i \nabla\psi\left(e_i^2\right)\mathbf{d} \qquad \mathbf{H}_a = \sum_{i=1}^{P} \nabla\psi\left(e_i^2\right)\mathbf{d}\mathbf{d}^T \quad (3.79)$$

$$\mathbf{h}_\mathbf{z} = \lambda_\mathbf{z} \sum_{(i,j)\in\mathcal{E}} \kappa_{ij}\, z_{ij} \nabla\varrho_\mathbf{z}\left(z_{ij}^2\right)\mathbf{l}_{ij} \qquad \mathbf{H}_\mathbf{z} = \lambda_\mathbf{z} \sum_{(i,j)\in\mathcal{E}} \kappa_{ij} \nabla\varrho_\mathbf{z}\left(z_{ij}^2\right)\mathbf{l}_{ij}\mathbf{l}_{ij}^T \ , \quad (3.80)$$

where $\mathbf{z}$ is either $\mathbf{u}$, $\mathbf{v}$ or $\mathbf{w}$, and:

$$z_{ij} = \mathbf{z}_{(i)}^c - \mathbf{z}_{(j)}^c \quad (3.81)$$

$$\mathbf{l}_{ij} = \left(\mathbf{1}_i^{(n)} - \mathbf{1}_j^{(n)}\right) \quad (3.82)$$

$$e_i = \mathscr{A}(\mathbf{x}_i;\mathbf{w}^c) - \mathscr{I} \circ \mathscr{W}(\mathbf{x}_i;\mathbf{u}^c,\mathbf{v}^c) \quad (3.83)$$

$$\mathbf{d} = \begin{bmatrix} -\nabla_\mathbf{u}\mathscr{W}(\mathbf{x}_i;\mathbf{u}^c,\mathbf{v}^c)\nabla_\mathbf{x}\mathscr{I}(\vec{\mathbf{x}}_i) \\ -\nabla_\mathbf{v}\mathscr{W}(\mathbf{x}_i;\mathbf{u}^c,\mathbf{v}^c)\nabla_\mathbf{x}\mathscr{I}(\vec{\mathbf{x}}_i) \\ \nabla_\mathbf{w}\mathscr{M}(\mathbf{x}_i;\mathbf{w}^c) \end{bmatrix}. \quad (3.84)$$

Algorithm 3 Optimisation Regime for Solving the Pairwise MAP Problem

Require: $\{\mathscr{I}^0, \mathbf{s}^0\}$ (template), $\mathscr{I}$ (image to fit to), $\mathbf{q} = [\mathbf{u}; \mathbf{v}; \mathbf{w}]$ (initial deformations), N_i (number of iterations) and ϵ (convergence tolerance)

Compute $\nabla_{\mathbf{x}}\mathscr{I}$, $\nabla_{\mathbf{u}}\mathscr{W}$, $\nabla_{\mathbf{v}}\mathscr{W}$ and $\nabla_{\mathbf{w}}\mathscr{M}$ {Equation (A.5)}
for $i = 1$ to N_i **do**
  Compute residuals $\{e_i\}_{i=1}^{P}$ and $\{u_{ij}, v_{ij}, w_{ij}\}_{(i,j)\in\mathcal{E}}$ {Equations (3.83) and (3.81)}
  Compute robust weights $\{\nabla\psi\left(e_i^2\right)\}_{i=1}^{P}$ and $\{\nabla\varrho_{\mathbf{u}}(u_{ij}^2), \nabla\varrho_{\mathbf{v}}(v_{ij}^2), \nabla\varrho_{\mathbf{w}}(w_{ij}^2)\}_{(i,j)\in\mathcal{E}}$
  Compute $\{\mathbf{h}_z, \mathbf{H}_z\}_{z=\{a,u,v,w\}}$ {Equations (3.79) and (3.80)}
  Compute parameter updates $\Delta\mathbf{q}$ {Equation (3.85)}
  Update current parameters $\mathbf{q} \leftarrow \mathbf{q} + \Delta\mathbf{q}$
  if $\|\Delta\mathbf{q}\|^2 < \epsilon$ **then**
    break. {Convergence achieved}
  end if
end for
return $\mathbf{u}, \mathbf{v}, \mathbf{w}$

With these linearisations, the error function is now in quadratic form. As such, differentiating with respect to the deformation updates $\Delta\mathbf{q}$ and equating to zero, the solution for the deformation updates takes the form:

$$\Delta\mathbf{q} = -\left(\mathbf{H}_a + \begin{bmatrix}\mathbf{H}_{\mathbf{u}} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{H}_{\mathbf{v}} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{H}_{\mathbf{w}}\end{bmatrix}\right)^{-1}\left(\mathbf{h}_a + \begin{bmatrix}\mathbf{h}_{\mathbf{u}}\\ \mathbf{h}_{\mathbf{v}}\\ \mathbf{h}_{\mathbf{w}}\end{bmatrix}\right) \tag{3.85}$$

Here, all $\mathbf{0}$ matrices are of size $(n \times n)$. A summary of the deformation optimisation procedure is outlined in Algorithm 3.

As a final note on the MML/MAP procedure, one notices that it is intuitively similar to typical methods for groupwise correspondence learning (see [9; 133], for example), where updates of the model and correspondences are interleaved, maximising the correspondences for a fixed model and *vice-versa*. Although no proof of convergence is currently available for such approaches, there have been no reports of algorithmic divergence in any publications on the groupwise method. In all experiments, presented in the next section, it was found that in no case did the procedure diverge.

3.6 Empirical Validation

In this chapter so far, a pairwise correspondence learning approach has been outlined, which leverages on the assumption of (piecewise) smooth spatial and appearance deformations. Utilising the approach of hierarchical Bayesian priors, through the MML/MAP method, an alternating procedure was proposed whereby all free variables in the problem, including those relating to regularising weights, can be tuned automatically for each image. In this section, the efficacy of the pairwise method is evaluated on a database of human faces.

The database used for all experiments in this section is described in Section 3.6.1. The performance of the pairwise method is then evaluated on partitions of this database, namely:

- A person specific database (see Section 3.6.2), where the inter-class variability stems from variations in the subject's pose, expression and external lighting effects.
- A pose specific database (see Section 3.6.3), where pose, expression and external lighting variations are kept to a minimum, with the main sources of variability stemming from inter-subject variations, such as facial hair.
- A generic face database (see Section 3.6.4), where variations now include inter-subject variabilities as well as those stemming from identity.

3.6.1 The IMM Face Database

For all experiments in this Section and indeed this thesis, the IMM Face database [89] is used exclusively. It consists of 240 images of 40 individuals, each exhibiting variations in pose, expression and lighting. The types of variations for each individual in the database are exemplified in Plot (a) of Figure 3.4. Note that the sources of variation are isolated in all but the last image, where the subjects exhibit random expression and pose. The first image of every subject exhibits a frontal pose with a neutral expression. In the second image, the subjects are again in a frontal pose, this time with a smiling expression. Pose variations are encoded in the third and fourth images, where the subject varies his/her head yaw angle between $\pm 30^o$, keeping a neutral expression. The effects of lighting variation are captured in the fourth image, applying directional light on the subjects' faces. In the sixth image, the subjects exhibit free variations in both pose and expression.

A 58-point markup is supplied with the database, allowing the performance of automatic correspondence learning methods to be evaluated quantitatively. Some examples of this annotation are shown in Plot (b) of Figure 3.4. Few other existing databases provide such annotations. Notable amongst these are the XM2VTS [85] database, with a 68-point markup[6], and the AR Face database [80], with a 22-point markup[7]. However, the XM2VTS database exhibits inter-subject variabilities only, with no variations in pose, lighting or expression between subjects in the database. The AR Face database exhibits an excellent range of variabilities, including occlusions due to clothing and glasses. However, the 22-point markup may be too sparse to capture the spatial variabilities of the human face.

3.6.2 Person Specific Databases

In this section, the ability of the pairwise approach to learn correspondences across a database of the same subject is evaluated. For this, the IMM Face database is partitioned into 40 groups of images, each containing only one subject. The pairwise learning algorithm is then applied to each group separately, setting the template image $\mathscr{I}^0$ to be the first image in the group (i.e. the image where the subject is in a frontal pose with a neutral expression).

There are two aspects of the pairwise procedure that must be evaluated here. The first is how well the (piecewise) smooth assumptions regarding the spatial and appearance deformations model the true intra-class variabilities. For this, the correspondences in each image are

[6] http://www.isbe.man.ac.uk/~bim/data/xm2vts/xm2vts_markup.html
[7] http://www.isbe.man.ac.uk/~bim/data/tarfd_markup/tarfd_markup.html

(a)

(b)

Figure 3.4: The IMM Face database. **(a):** example variabilities within the database. **(b):** examples of the 58-point markup of the database.

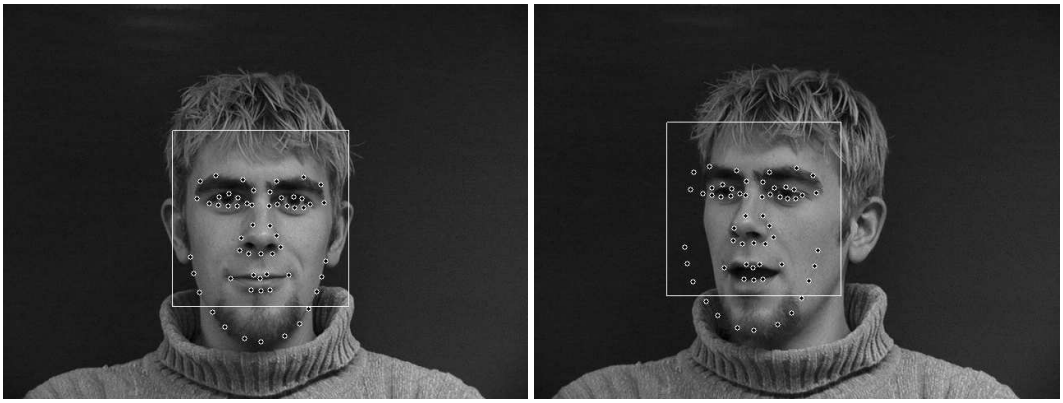


Figure 3.5: An example of correspondence initialisation using an affine transformation between detected bounding boxes.

set to their optimal locations (i.e. the manual annotations). The EM-algorithm (steps 4 to 6 in Algorithm 2) is then iterated until convergence, followed by an optimisation of the deformations as outlined in Algorithm 3. This way, the hyperparameters $\{\tau, \alpha, \beta, \gamma\}$ are computed using the correct correspondences, and the amount by which the shape then deforms from its optimal initial values reflects how well the (piecewise) smooth assumption models intra-class variabilities, such as pose, expression and lighting. It should be noted here, that although the correspondences are initialised at their true location, the appearance deformations cannot be, since no manual labels for these are available. Furthermore, since the EM-algorithm guarantees convergence only to a local optimum, the values of the hyperparameters used in optimising the deformations may still be sub-optimal. As such, the results of experiments on this aspect of the pairwise procedure may underestimate the true capacity of (piecewise) smooth deformations in modelling intra-subject variabilities.

The second aspect to be evaluated is the sensitivity of the pairwise procedure to initialisation. For this, the initial correspondence estimate is obtained by applying an affine transformation between the template image and all others in the database. The affine transformations are computed from pairs of bounding boxes, one in the template and one in another image, which are found by applying a face detector over the whole database. In this thesis, OpenCV's[8] face detector is utilised for this purpose, which implements the Viola and Jones method [135]. An example of the initial correspondence estimate obtained from this affine transformation is illustrated in Figure 3.5. Since the initial estimates of the correspondences can be far from their optimal locations, experiments on this aspect of the pairwise procedure are performed on a Gaussian pyramid of three levels.

In Figure 3.6, the convergence of the hyperparameters for the optimally initialised correspondences is shown for a number of hyperparameter initialisations. In these experiments, the robust variant of the image likelihood was used, as were the spatial and appearance deformations (i.e. piecewise smooth deformations are assumed). As described in Section 3.3.2, the priors are penalised using the regularised L_1 norm in Equation (3.20), with ϵ set to 0.0001. The same robust penaliser is used in the image likelihood since it avoids choosing a robust param-

[8] `http://sourceforge.net/projects/opencvlibrary`

Table 3.1: Person specific experiments with manual initialisation

Experiment	Appearance Deformation	Image Likelihood	Deformation Prior
(a)	✓	robust	robust
(b)	✓	robust	non-robust
(c)	✓	non-robust	non-robust
(d)	×	robust	robust
(e)	×	robust	non-robust
(f)	×	non-robust	non-robust

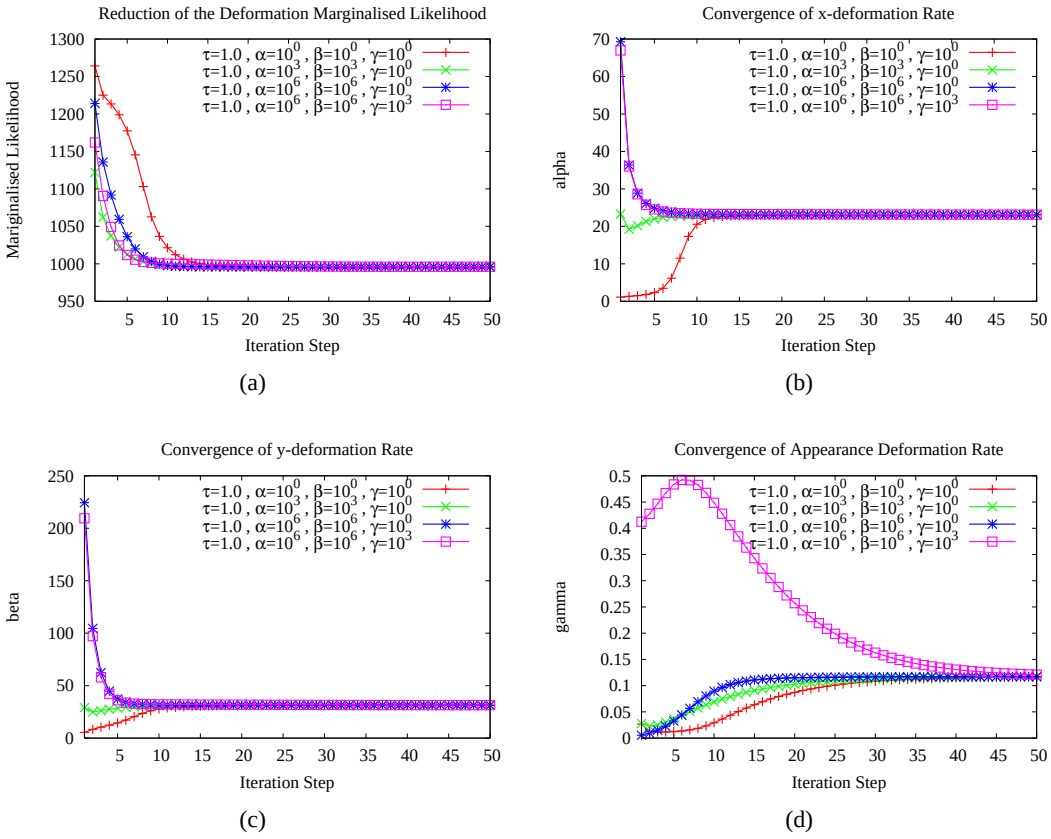


Figure 3.6: Illustration of the convergence of the hyperparameters using optimally initialised correspondence at four initial settings of of the hyperparameters (shown in the legend). **(a):** the reduction of the MML error in Equation (3.58). **(b), (c) and (d):** the evolution of τ, α and β is shown respectively, throughout the EM-algorithm's iterations.

eter, which is required by most other robust penalisers[9]. Although the choice of the L_1 norm for penalising appearance differences may be suboptimal, its successful application has been previously demonstrated in numerous variational optical flow methods (see [22; 23; 110], for example). The local convergence property of the EM-algorithm is illustrated in Plot (a), where the error in Equation (3.58) is shown to monotonically decrease throughout the iterations. The evolution of the hyperparameter values throughout the procedure is shown in Plots (b), (c) and (d). In this particular instance the optimisation finds the same solution for each of the four trials. The same experiment performed on different images also exhibited similar behaviour. As such, although the EM-algorithm affords only a local optimum of the error function, experiments indicate that the procedure is fairly insensitive to the initial choice of hyperparameters when the shape correspondences are optimally initialised.

Results of applying Algorithm 3 to the deformations, starting with optimal correspondence and hyperparameter estimates are shown in Figure 3.7, where results from all subjects have been combined. Here, six experiments were performed for each subject, outlined in Table 3.1. Notice that experiments (d) to (f) do not use the appearance deformation model described in Section 3.3.1. Instead, the likelihood is evaluated by directly comparing the cropped image with the template. In experiments (b), (c), (e) and (f), non-robust priors are used to compare the performance of the assumption of strictly smooth deformations against the assumption of piecewise smooth deformations. Finally, in experiments (c) and (f), the non-robust likelihood is utilised to evaluate the applicability of robust penalisers in the case of person specific correspondence learning. Note that all the derivations presented in this chapter can be applied to the non-robust case by replacing the derivative of the robust penaliser with a value of unity, wherever it occurs.

Comparing the results of experiment (a) with (b), and (d) with (e), it appears that the use of a robust penaliser in the priors, which implies piecewise smooth deformation, has little effect on the convergence accuracy of the method on this database. Comparing these with the results for experiments (c) and (f), where a non-robust likelihood is used, one notices a deterioration in accuracy when the non-robust formulation is used. These results are somewhat counterintuitive, since the variation in appearance between instances of the same subject is expected to be small, but the variation in shape, large, due to the varying pose exhibited in the database. However, the database exhibits varying lighting and expressions, both of which induce significant appearance differences. It also appears that, although the variation in shape between instances of the same subject is large, the types of exhibited deformations are smooth. Comparing the results of experiments (a), (b) and (c) with those of experiments (d), (e) and (f), it is clear that in all cases, the utility of the appearance deformation is well justified as the performance is improved when it is deployed, albeit only by a small amount. It is expected that the appearance model is particularly useful on the images where directional lighting is applied to the subjects. From these results, it can be concluded that the model which best approximates the generative properties of a person specific model is that which utilises a robust likelihood penaliser and an appearance deformation model, whilst the utility of a robust penaliser in the deformation priors has little effect on the global optimum of the formulation.

A final outcome of these results is that the correspondence errors are not spread equally

[9]The regularised L_1 norm is fairly insensitive to the choice of ϵ that is applied for numerical reasons only.

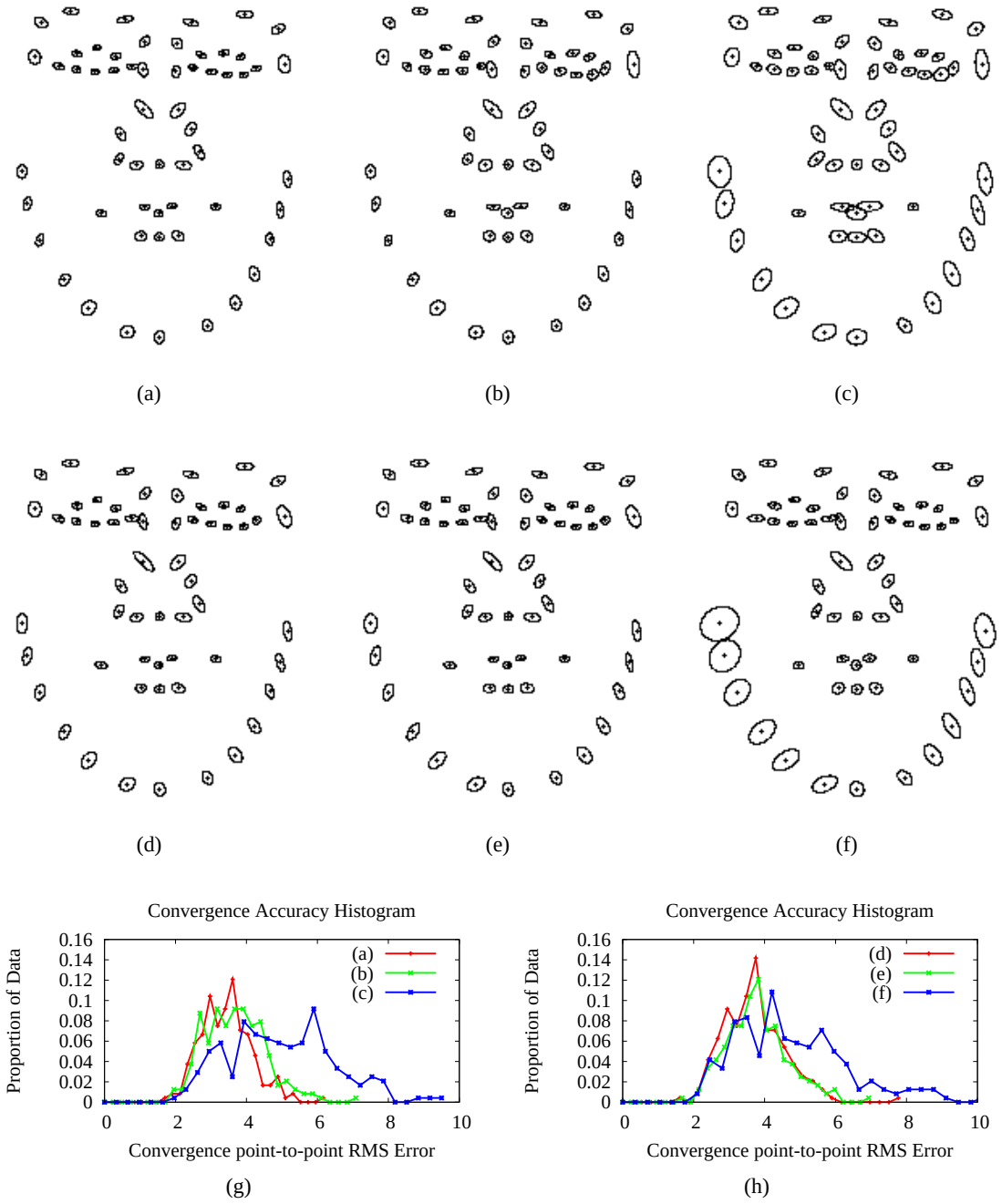


Figure 3.7: Performance of the pairwise method on person specific databases starting from optimal correspondences. **(a) to (f):** one standard deviation ellipses of converged per-point error for every landmark in experiments (a) to (f). **(g):** Accuracy histograms of experiments (a), (b) and (c). **(h):** Accuracy histograms of experiments (d), (e) and (f). Note that error is defined as the point-to-point RMS error, measured from manual annotations.

amongst the landmarks. Those exhibiting the largest errors in all six experiments were those around the nose and jaw line. Due to the significant pose variations within the database, the nose actually occludes part of the cheek in some images. The result of this is that the shape deforms to accommodate the difference in appearance in this region from that of the template. Also, since the template is defined in the canonical frame, there exists a depth ambiguity regarding where the landmarks on the upper jaw line are actually situated. As such, when fitting to images with extreme pose difference from the template, in some cases, the pairwise method fails to extract the correct locations.

The results of applying the optimal parameterisation (experiment (a)) to the bounding box initialised correspondences is shown in Figure 3.8. It is clear that despite performing optimisation on a Gaussian Pyramid, the problem of converging to a local optimum is still prevalent, observed through the significant deterioration in the method's performance compared to its optimally initialised counterpart. Nonetheless, the models built using this approach may still exhibit some utility for face fitting.

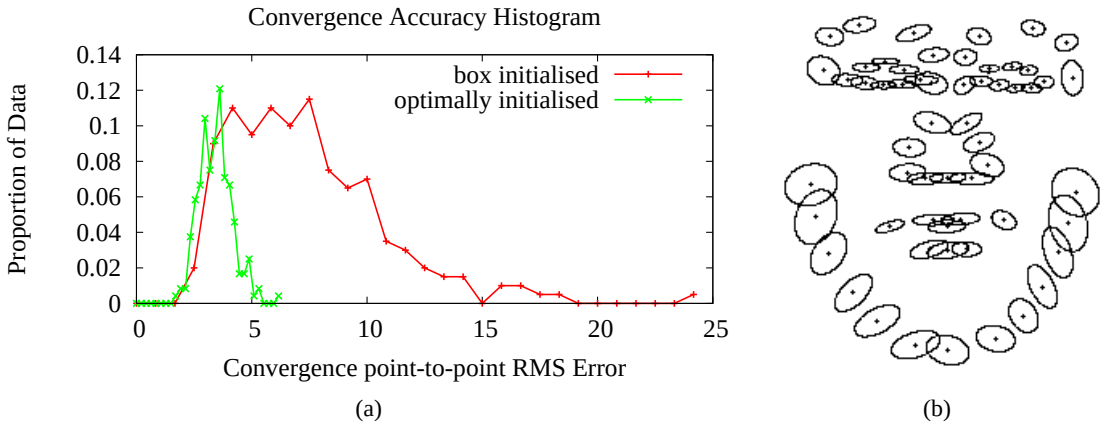


Figure 3.8: Performance of the pairwise method on person specific databases using bounding box initialisation. **(a):** Accuracy histograms at convergence. Note that the sampling rate of the optimally initialised histogram is higher than that of the box initialised histogram, explaining the apparent differences in area under the histograms. **(b):** one standard deviation error ellipses for each landmark.

A qualitative evaluation of the found correspondences in the experiments described above can be obtained by building a linear model of shape and appearance over the database and inspecting the resulting reconstructions. Some examples of this are shown in Figure 3.9. Here, the model built using manual annotations, those from experiment (a) and (f) with optimal initialisation, and those from experiment (a) with a bounding box initialisation, are varied between ± 3 standard deviations of their first mode of combined appearance variation. Inspecting the results from automatic correspondence learning, one notices that although some differences in shape from the manual model can be observed, the appearance reconstructions are *crisp* with no significant ghosting or blurring effects. This is the case even for the bounding box initialised method, which was shown quantitatively to attain much poorer correspondences with respect to manual annotations.

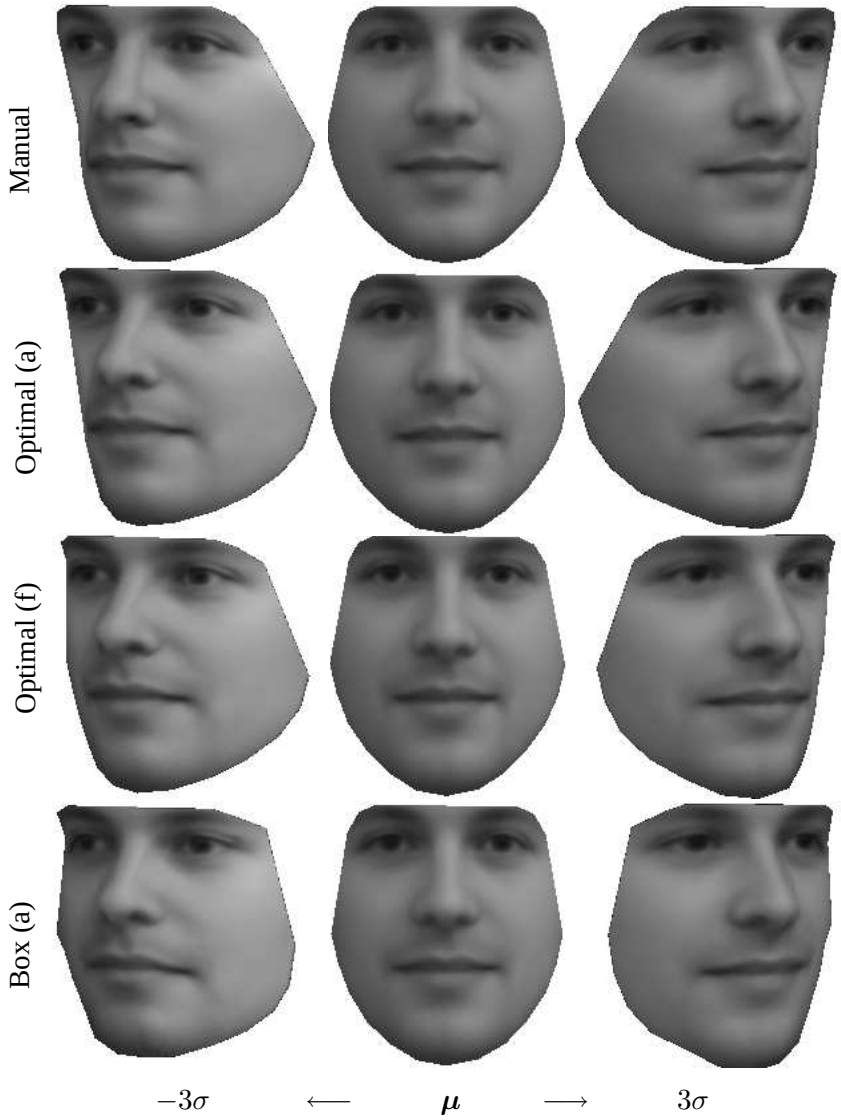


Figure 3.9: Reconstruction results of intra-person pairwise learning. The model was built using all subjects in the database, with the variations shown corresponding the the first mode of combined appearance variation.

3.6.3 Pose Specific Database

In this section, the ability of the pairwise approach to learn correspondences across a database of varying subjects with fixed pose, expression and lighting is evaluated. For this, only the first image (frontal pose with neutral expression) of each subject in the IMM Face database is utilised. To evaluate the effects of different templates on the performance of the method, experiments were conducted using four separate templates: a male subject with a beard, a female subject, a male subject with no facial hair, and a male subject with a moustache. The four chosen templates are shown in Plot (a) of Figure 3.10. For each chosen template, the same experiments were conducted as in the person specific case, outlined in Table 3.1.

The results of experiments (a) to (f) using the first template with optimally initialised correspondences and hyperparameters are presented in Figure 3.11. Comparing the results for experiments (a) with (b), and (d) with (e), it can be seen that the use of a robust deformation prior has the capacity to improve fitting performance. However, the improvements here are marginal, especially in experiments that do not utilise an appearance model (i.e. experiments (d) and (e)). As such, compared to the person specific case, pose specific databases appear to exhibit more discontinuities in deformation, although their amount is fairly constrained. However, comparing the results from experiments (b) with (c), and (e) with (f), a significant deterioration can be observed when a non-robust likelihood is utilised. This is to be expected, since the difference between the template and image contains localised regions with large errors, stemming from such sources as facial hair and general difference in appearance between individuals. Finally, comparing the results of experiment (a) and (d), it is clear that the use of an appearance model in this case affords an improvement in the accuracy of found correspondences. However, this result is not repeated in the other experiments. A possible cause for this might be due to the initialisation procedure used for the appearance model parameters. In all experiments, the appearance model is initialised to zero. As such, in the early stages of optimisation, the effects of appearance difference dominate the shape updates, leading to significant perturbation of the correspondences. In some cases, the procedure may settle in local minima. This characteristic may not have been exhibited in the person specific case, since intra-person appearance differences are fairly constrained. From these experiments, it can be concluded that the optimal parameterisation for a pose specific database, exhibiting variations in identity, is one that utilises robust penalisers in the likelihood and prior, as well as allowing the template's appearance to deform along with its shape.

The results of using the other three templates are similar, where the per-point accuracies of each in experiment (a) are shown in Plot (b) of Figure 3.10, along with the accuracy histograms of all four templates. Notice that the landmarks that exhibit the largest errors are those around the extremities of the model, namely the eyebrows and the chin. This pattern is significantly different from the person specific case, where the largest errors occur around the upper jaw line and the region around the noise. Since the main source of variability in this database is due to identity, the pattern of errors here can be explained as the effects of template-image mismatch. The variations around the chin and mouth are the result of some subjects exhibiting beards and/or moustaches. This causes the model to deform around this area. Differences in eyebrow thickness and shape are also prevalent within the database, leading to variations that can not be well accounted for by the appearance deformation function.

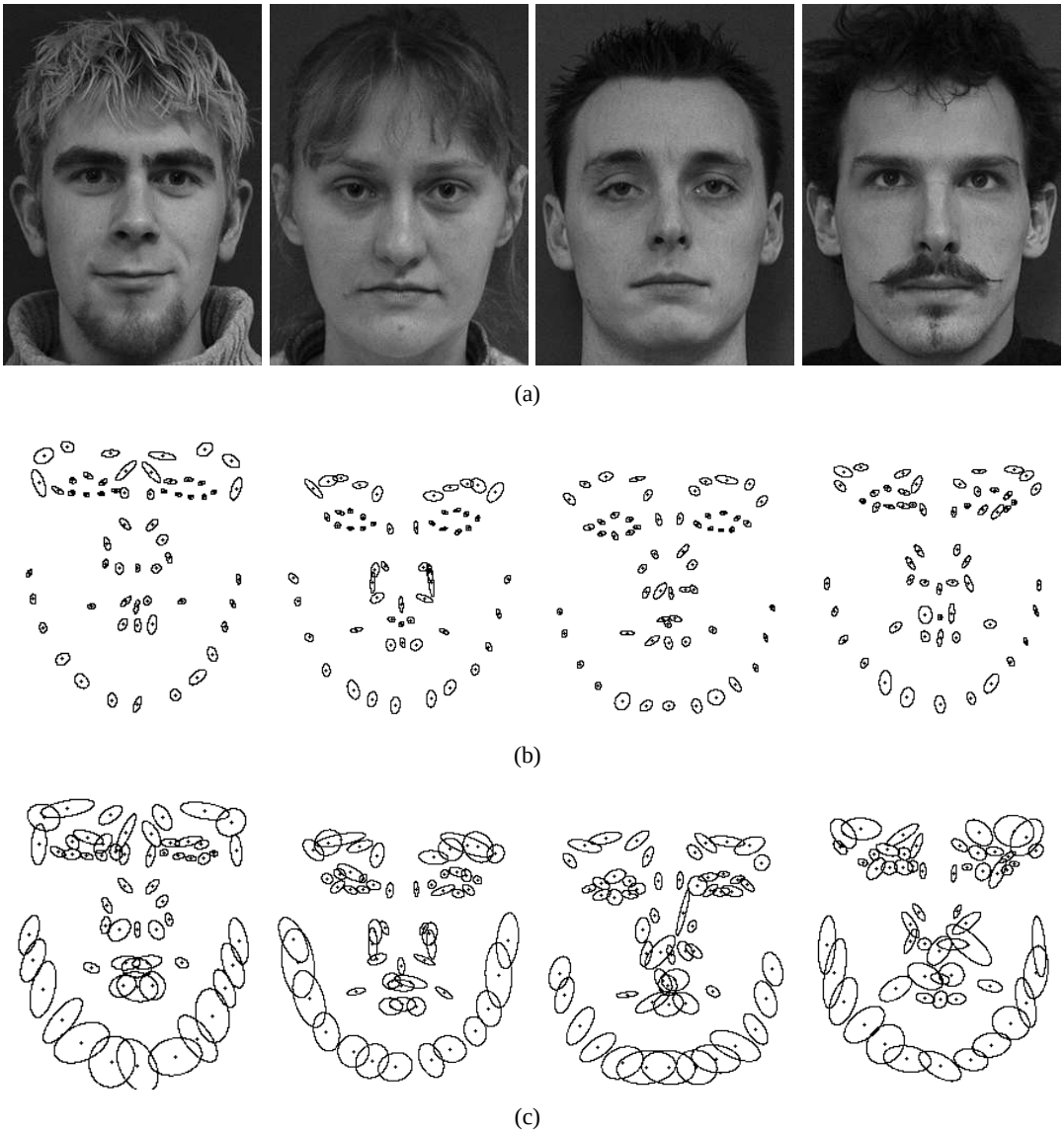


Figure 3.10: (a): The four chosen templates for the pose specific experiments. **(b):** the one standard deviation ellipse of converged per-point error for every landmark, starting from optimal annotations. **(c):** converged per-point error using bounding box initialisation.

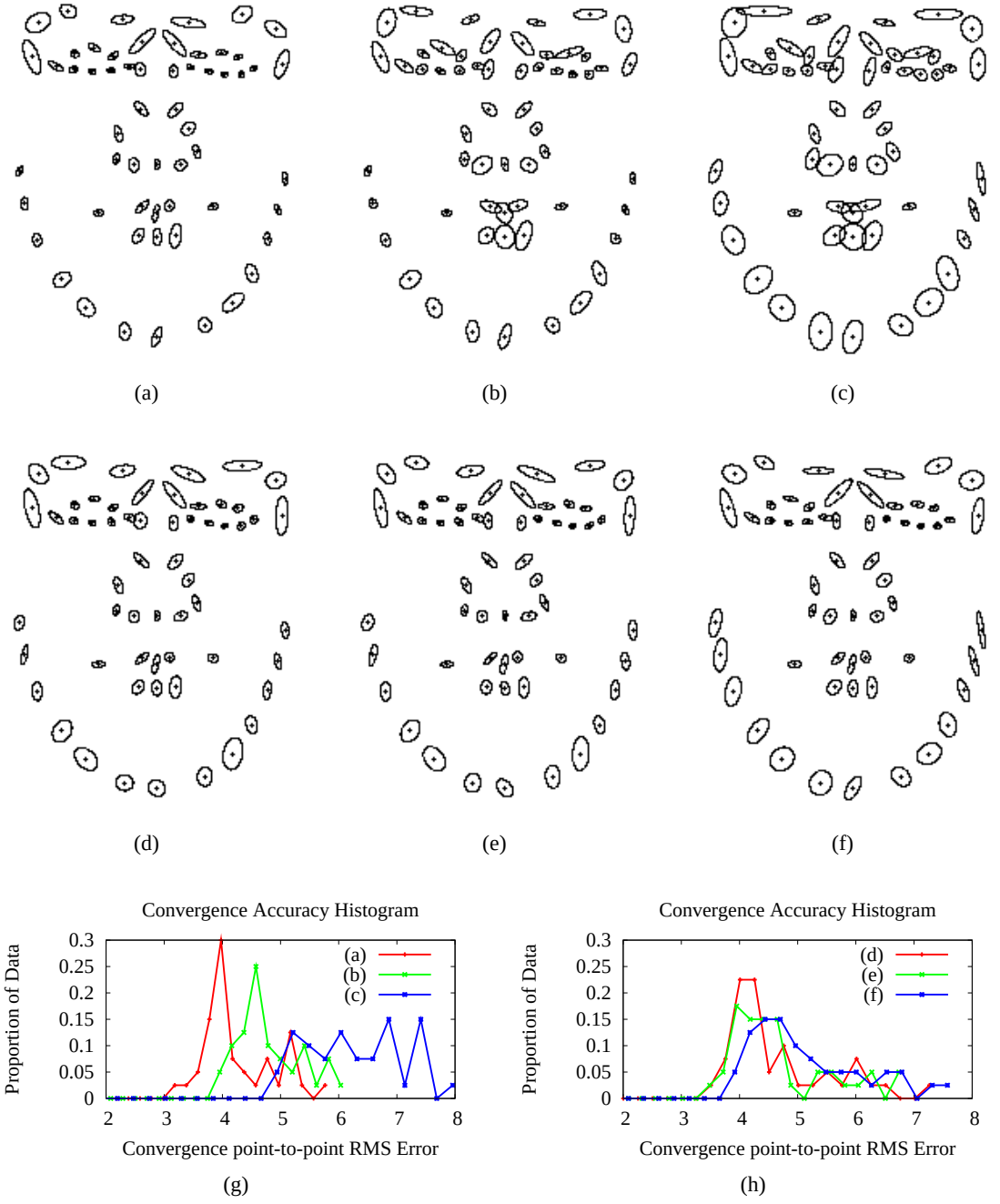


Figure 3.11: Performance of the pairwise method on a pose specific databases, starting from optimal correspondences. **(a) to (f):** one standard deviation ellipses of converged per-point error for every landmark in experiments (a) to (f). **(g):** Accuracy histograms of experiments (a), (b) and (c). **(h):** Accuracy histograms of experiments (d), (e) and (f). Note that error is defined as the point-to-point RMS error, measured from manual annotations.

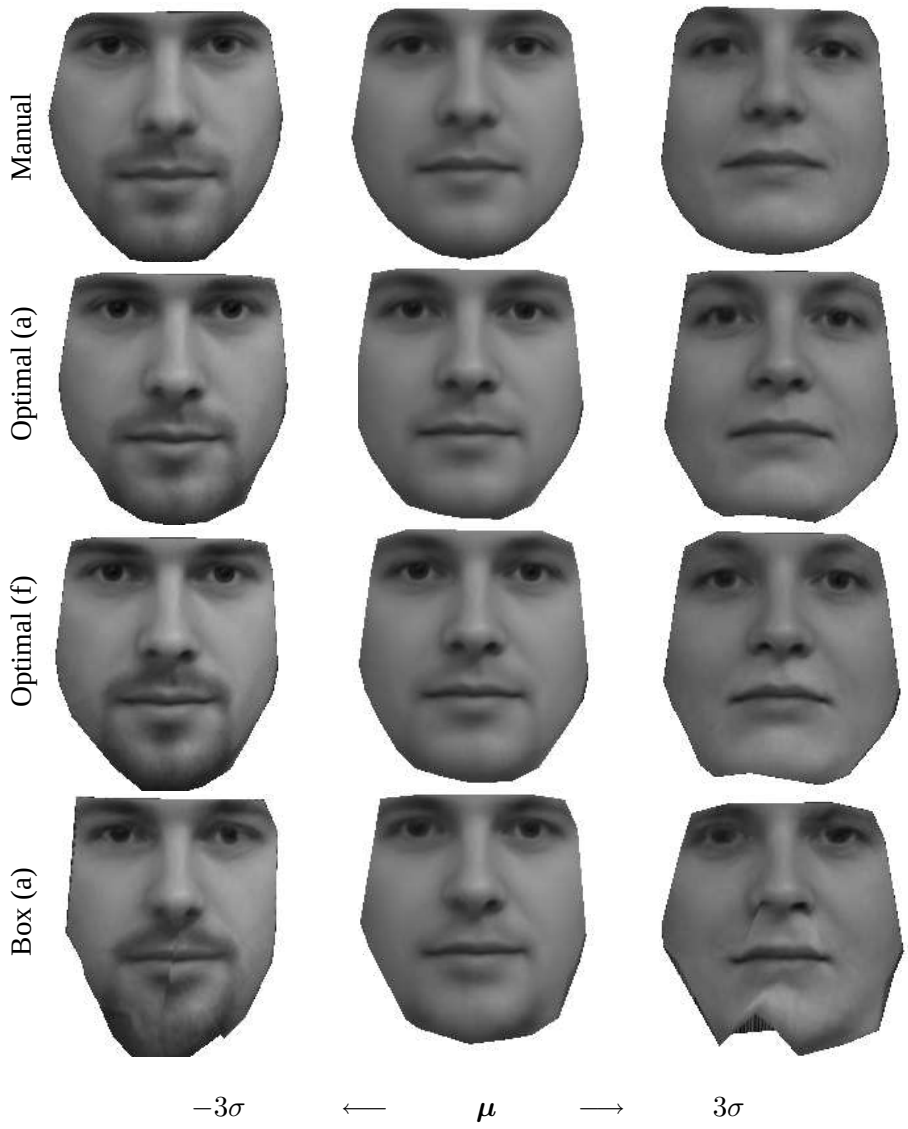


Figure 3.12: Reconstruction results of inter-person pairwise learning. The model was built using all subjects in the database, with the variations shown corresponding the the first mode of combined appearance variation.

The results of applying the optimal parameterisation (experiment (a)) to the detector initialised correspondences are shown in Figure 3.10 for each of the templates. As with the person specific case, it is clear that despite performing optimisation on a Gaussian Pyramid, the approach is highly sensitive to initialisation, terminating in local minima in a large proportion of the images. Reconstructions from models built using the manual and automatic correspondences for some of the experiments are also shown in Figure 3.12. As with the subject specific case in the previous section, reconstructions using results of the optimally initialised experiments exhibit good properties. Although some differences in shape from the manual model can be observed, there are no significant ghosting or blurring effects. However, the reconstruction results of the box initialised model are far from satisfactory. Here, significant ghosting and blurring effects can be observed as well as highly unrealistic shape contortions.

3.6.4 Generic Person Database

As a final set of experiments, the ability of the pairwise approach to learn correspondences across a generic person database with varying identity, pose, expression and lighting, is evaluated. For this, the whole IMM Face database is used, choosing the first image in the database as a template (i.e. the results from the previous section suggest that the choice of template has only a marginal effect on accuracy compared to the parameterisation of the model). Again, the same experiments were performed as in the person specific case, outlined in Table 3.1.

The results of experiments (a) to (f), using the optimally initialised correspondences and hyperparameters are presented in Figure 3.13. In contrast to the results in the previous sections, here a clear trend of accuracy improvement can be observed as the likelihood and priors are robustified as well as when the appearance deformation model is used. In fact, the improvement in accuracy attained by utilising an appearance deformation model is quite marked compared to those in the previous section. Examining the per-point accuracy images for experiments (a), (b) and (c), one notices that the pattern of errors is a combination of the patterns for the person specific and constrained generic person cases. This is to be expected however, since the main difference between the experiments here and those in the previous section, is the inclusion of pose, expression and lighting variability into the database. As such, the deficiency of the pairwise method due both to intrinsic and extrinsic variabilities are compounded when both sources of variations are present. However, for experiments (d), (e) and (f), the errors are much larger than a simple combination of the errors in the two preceding sections. This can be attributed to the poor modelling capacity of the template when both intrinsic and extrinsic sources of variability are present. It appears, therefore, that when variations in pose, expression, lighting and identity are present within the database, the utility of an appearance deformation model is crucial to attaining good results.

Experiments on the bounding box initialised correspondences were not conducted on this database. However, the performance of the pairwise method in this case can be expected to exhibit the same difficulties regarding local minima as those discussed in the person specific and pose specific cases.

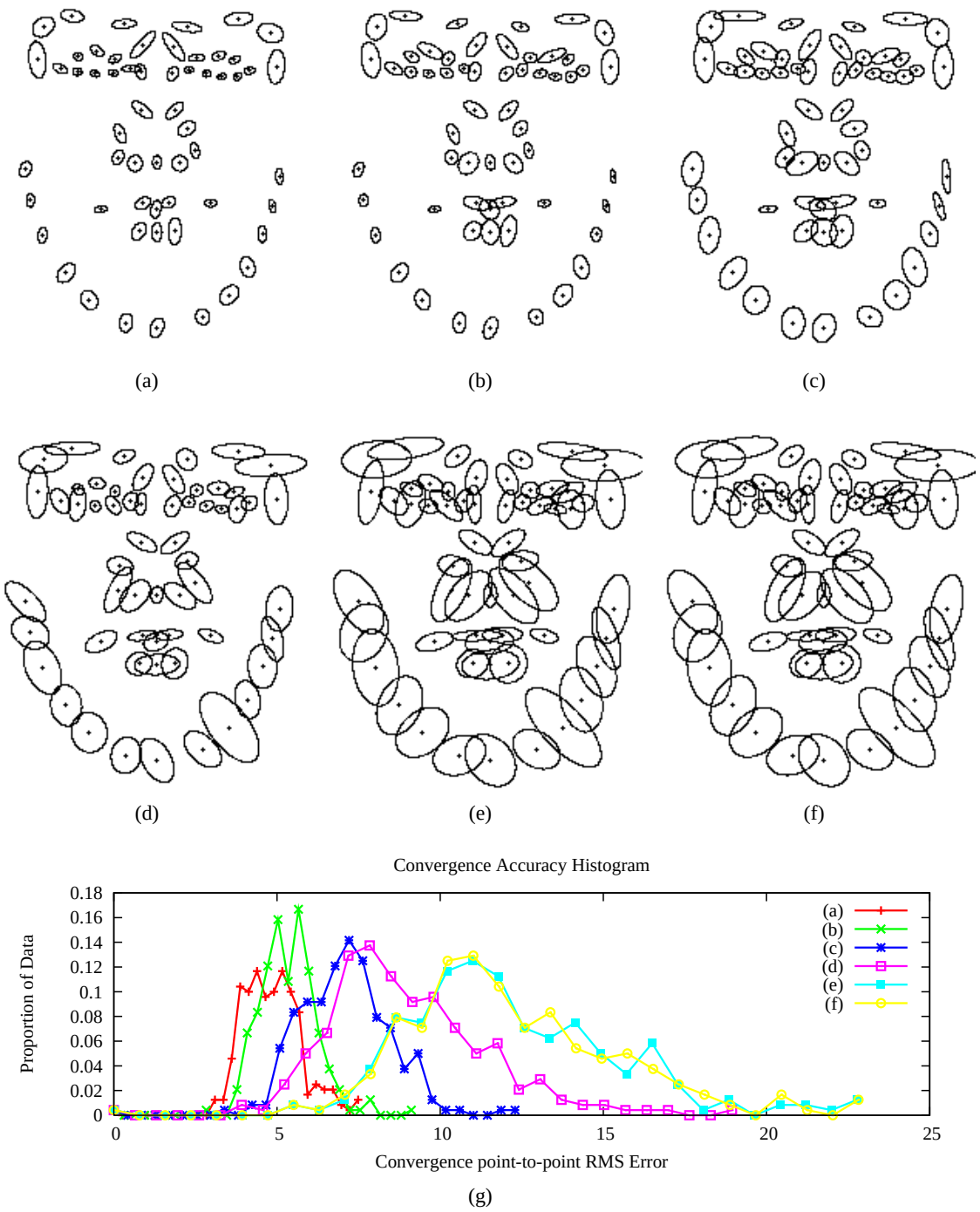


Figure 3.13: Performance of the pairwise method on a generic person database, starting from optimal correspondences, using the first image in the database as a template. **(a) to (f):** the 1 standard deviation ellipse of converged point-to-point error for every landmark. **(a) to (c):** results of using an appearance deformable template. **(d) to (f):** results of using a purely spatially deformable template. **(a) and (d):** results of using robust penalisers for both the likelihood and priors. **(b) and (e):** results of using a robust penaliser for the likelihood only. **(c) and (f):** results of using non-robust likelihood and priors. **(g):** accuracy histogram plots of the six cases, defining error as the point-to-point RMS error, measured from the manual annotations.

3.7 Conclusion

In this chapter, a novel automatic correspondence learning approach has been proposed that learns the optimal deformations from a pre-annotated template image to an un-annotated image under the constraint of (piecewise) smooth deformations, both in the spatial and appearance domains. The approach is formulated within a Bayesian framework, utilising inference with hierarchical priors to allow all free variables within the problem to be tuned automatically. Using an EM procedure to maximise the data log-likelihood, the procedure guarantees a local solution for each linearisation of the cropped image and robust penalisers. Compared to existing methods, the pairwise approach described here exhibits a number of advantages:

- A formal description of optimality by virtue of its formulation within a Bayesian framework.
- It affords automatic tuning of the regularisation weight in the regularised data fitting analogy.
- The approach allows extensions to take into account extra prior information about the visual object of interest.

Through experiments on a database of human faces, the strengths and weaknesses of the approach have been evaluated. From these, it was seen that the use of the appearance deformation model, piecewise smooth deformations and a robust image matching function, gave significant performance improvements, especially in cases where the database exhibits large amounts of variabilities. However, it was also demonstrated that the procedure is sensitive to initialisation.

Improvements to the proposed pairwise approach can be made on two fronts. The first is to integrate domain knowledge about the database at hand. This can take the form of priors on the correspondences, which can be integrated elegantly into the proposed formulation. For example, for a dataset of an image sequence, the conditional dependence between correspondences in consecutive images should be accounted for, possibly assuming (piecewise) smooth transitions between images. Another example is the case where there exists multiview-stereo images in the database, in which case dependencies between the multiview-images can be incorporated into the formulation. The Bayesian formulation of the pairwise approach allows these types of domain knowledge to be integrated in a formal manner. When a small number of correspondences across the images is available, this information can also be integrated as a prior. It has been shown in [101], that increasing the number of features in deformable model fitting has the effect of smoothing the optimisation's error terrain, thereby reducing the likelihood of the procedure terminating in local minima. As sensitivity to local minima is a weakness of the proposed pairwise method, investigations into the effects of utilising multiple priors and likelihood terms constitute a good possibility for future work.

The second area in which improvements may be made is in the assumptions made regarding the distributions of the visual object. In cases where additional knowledge about the visual object of interest is available, a more representative distribution function modelling the object will result in a more constrained problem, and hence a more compact solution. For example, the number and placement of anchor points for the appearance deformation may be suboptimal.

Another is the type of robust penaliser used in the image likelihood. These modifications are less attractive, however, since the optimal choices for each requires a hit-and-miss approach.

Finally, the general method for automatic correspondence learning proposed in this chapter can be adapted to the problem of groupwise correspondence learning. An example of how this may be achieved is presented in Appendix B. In the method presented there, appearance and shape deformations are modelled as a linear model, which may better suit the types of objects often learnt in correspondence learning than the model presented in this chapter. Furthermore, the MML criterion is better approximated, since linearisation is required only for the cropped image, rather than the robust penalisers as well. Due to time constraints, this method has not been implemented at the time of this writing, however it constitutes a strong possibility for directions of future work.

Chapter 4

Iterative-Discriminative Fitting

He turned the power to the have-nots.
And then came the shot!

Rage Against the Machine

As discussed in Section 2, there has been a large amount of work done recently in an attempt to improve the performance of LDM fitting. However, most methods address some of the fitting goals at the expense of others. The project-out inverse compositional method [83], for example, boasts extremely rapid fitting at the expense of poor generalisability. As such, despite the significant advances so far, accurate, efficient, reliable, automatic and applicable fitting is still an open problem.

In this chapter, a novel discriminative fitting paradigm is outlined, which presents a significant step forward in addressing all of the aforementioned goals. The main idea is to reduce the error bounds over the data, rather than the typical least squares criterion. Combined with an iterative scheme, all samples in the training set are guided towards their solution, placing a higher priority on samples with large errors. As the objective in the discriminative learning needs only be partially satisfied at each iteration, the approach allows simple regressors, which generally exhibit better generalisability than more complex ones, to be utilised. Generalisability is further promoted through a resampling process between iterations, artificially increasing the training set size. The approach is highly applicable, with no specific requirements placed on the model's parameterisation or the type of feature used to drive the fitting procedure.

The general problem of discriminative fitting is described in Section 4.1. Section 4.2 then describes the novel approach of iterative-discriminative fitting. Two implementations are then discussed in detail in Section 4.3, a linear approach and a nonlinear one. Extensions of the iterative-discriminative approach to robust fitting and background invariance are described in Sections 4.4 and 4.5 respectively. Section 4.6 concludes with an overview and a discussion on directions of future work. The experimental evaluation of the methods proposed in this chapter can be found in Chapter 5.

4.1 The Discriminative Fitting Problem

Discriminative learning, sometimes considered the opposite or an alternative to generative modelling, is an approach that attempts to directly learn the input-to-output mapping of a problem. No effort is wasted on the intermediate goal of explicitly modelling the underlying distributions of the variables and features in the problem. Instead, treating the problem as a black-box, the mapping function is adjusted purely to satisfy the function approximation

quantity, over which the performance of the method is later evaluated.

By virtue of the tight coupling between the training objective and evaluation criterion, discriminative methods have been shown to outperform generative methods in a number of problems. While the performance of generative methods relies heavily on how well the constructed distributions approximate that of the real problem, discriminative methods require only that the distributions of the training data sufficiently mimic that of the problem. As such, for constrained problems with sufficient training data, discriminative methods are a natural choice.

In the context of LDM fitting, the mapping function to be learnt is that which relates the observation obtained from a perturbed parameter setting to the optimal updates required to bring the model into alignment with the visual object in the image. Formally, for the given training set:

$$\{\mathscr{I}_i, \mathbf{p}_i, \Delta\mathbf{p}_i^*\}_{i=1}^{N_d}, \tag{4.1}$$

where $\mathbf{p}_i$ and $\Delta\mathbf{p}_i^*$ are the perturbed LDM parameters and their optimal updates, respectively, for the image $\mathscr{I}_i$, discriminative learning aims to find the update function (regressor) $\mathscr{U}$ that maps the feature extracted from the image at its current parameter settings, to the desired parameter update:

$$\Delta\mathbf{p}^* = \mathscr{U} \circ \mathscr{F}(\mathscr{I}; \mathbf{p}). \tag{4.2}$$

The feature extraction function:

$$\mathscr{F}(\mathscr{I}; \mathbf{p}) \colon \Re^{N_p} \rightarrow \Re^{N_f}, \tag{4.3}$$

where N_p and N_f represent the dimensionality of the LDM's parameters and the feature vector (observation), respectively, evaluated at the perturbed parameter settings $\mathbf{p}$, should be chosen such that the observation contains all the required information to allow $\mathscr{U}$ to perform an accurate estimation of the updates. Some examples of this function include the normalised raw appearance feature [29]:

$$\mathscr{F}(\mathscr{I}; \mathbf{p}) = \mathscr{N} \circ \mathscr{I} \circ \mathscr{W}(\mathbf{p}), \tag{4.4}$$

with $\mathscr{N}$ denoting the normalisation function, or the texture residual feature [4]:

$$\mathscr{F}(\mathscr{I}; \mathbf{p}) = \mathscr{A}(\mathbf{p}) - \mathscr{I} \circ \mathscr{W}(\mathbf{p}), \tag{4.5}$$

where $\mathscr{A}$ is an appearance generating function. In both Equation (4.4) and (4.5), $\mathscr{W}$ denotes the warping function (see Section (3.3.3)).

The training set of perturbed LDM parameters should be chosen to simulate the initialisation capacity of the detector, used to find the rough location of the visual object in the image. As such, the update functions are essentially trained on simulations of real fitting problems, in which case the performance of the updates on unseen images should approach that on the training set, if the simulated training cases are close approximations of the real problem.

There are a number of advantages of discriminative based approaches compared to generative based methods. Some of these are listed below.

- The regressors used to approximate the mapping function can be specialised to the problem. This can be achieved, for example, by utilising the same initialisation procedure

on the training set as in the test set. In this way, the distribution of the LDM parameters with respect to their optimal settings in the training images will closely approximate that of the test images well.

- The regressors are generally fixed, eliminating the requirement to recalculate the update model as with many generative methods (see [4; 12], for example). As such, when the type of regressor chosen is computationally cheap to evaluate, then an efficient fitting procedure results. Furthermore, the online computational cost is predictable due to the fixed update models used. This in turn allows simpler resource allocations for applications that utilise discriminative LDM fitting.

- Estimates of statistics regarding fitting performance, such as accuracy and frequency of convergence, can be directly attained from the training procedure without further evaluations on a test set. This allows the construction of likelihoods regarding the predicted perturbations, for use in further generative inference later if so desired.

- Generalisation can be directly integrated into the training procedure to reflect the confidence over the training set through the use of regularisation in learning.

- Flexibility regarding the trade-off between computational efficiency and accuracy can be directly designed into the learning procedure through the choice of the regressor function's complexity.

- The approach is applicable to many types of deformable objects and is not limited to specific warping functions, feature vectors or model parameterisation.

Although the advantages of using discriminative LDM fitting are numerous, some drawbacks of this approach have also been identified. Some of these include:

- The training procedure is generally much more computationally demanding and difficult to implement compared to that of generative methods.

- The best type of regressor and its coupling feature extractor function for a particular problem are difficult to deduce from domain knowledge. As such, a hit-and-miss approach must be utilised in general.

- Discriminative training generally requires a number of parameters that need to be either selected heuristically, or tuned to optimise some performance criterion over the training set.

As discussed in Section 2.3.3, in comparison to the generative approach, there exists only a few methods that tackle the problem of LDM fitting from a discriminative perspective. Perhaps this is because, despite exhibiting favourable properties, the drawbacks of the discriminative approach can be difficult to address. In the following sections, a novel discriminative procedure is proposed for the problem of LDM fitting, which addresses some of the aforementioned difficulties through a reformulation of the objective in discriminative learning, leveraging on the peculiarity of general fitting problems.

4.2 Iterative-Discriminative Fitting

With the discriminative formulation of LDM fitting described above, two questions naturally arise:

- Does there exist a mapping function that can accurately predict the updates over all legal states of the model for a given image?

- If one does exist, can it be evaluated efficiently?

The peculiarity of LDM fitting, as opposed to other problems commonly tackled by discriminative approaches, is that for many parameter settings, the feature vector will generally contain only a subset of the information required to perform accurate predictions directly. For example, the feature vector of a model perturbed in translation will not generally contain information regarding the boundary of the object in the direction opposite to that of the translation. As such, the predictions of the correct update must rely on the correlations between available information with that which is not. This relationship may be quite complex, requiring sophisticated regressors to predict it accurately. These complex regressors, in turn, may require significant computational resources to evaluate, negating one of the main advantages of discriminative methods. Furthermore, complex regressors usually exhibit poorer generalisability, leading to the requirement of a very large training set to cover, which in turn leads to slower training times.

To address these difficulties, an alternative discriminative framework will be considered, whereby a *set of weak regressors* are composed together to form a single *strong* predictor. Formally, rather than applying the parameter updates as in Equation (4.2), consider the parameter adaptation of the following form:

$$\mathbf{p} \leftarrow \mathbf{p} + \sum_{i=1}^{N_i} \Delta\mathbf{p}_i \quad \text{where} \quad \Delta\mathbf{p}_i = \mathscr{U}_i \circ \mathscr{F}\left(\mathscr{I}; \mathbf{p} + \sum_{j=1}^{i-1} \Delta\mathbf{p}_j\right). \tag{4.6}$$

Here, $\{\mathscr{U}_i\}_{i=1}^{N_i}$ is the set of fixed weak regressors. The intuition for utilising this particular form of regressor is as follows. Firstly, weak or simple regressors usually exhibit better generalisability than more complex ones. Secondly, by virtue of their sequential composition, observations of the image from a number of different parameter settings result in a richer information set used to make a prediction. This formulation, which will be referred to as the *iterative-discriminative* method in the remainder of this thesis, takes inspiration from boosting methods [47; 81; 134], where a set of weak learners are combined to form a strong one. If each weak learner can be efficiently evaluated, then an efficient fitting procedure may result. However, unlike boosting procedures where only the target of the mapping function is modified with each weak learner added, here the input (observation) is also modified to reflect the new distribution of samples around their optimum.

Algorithm 4 Iterative-Discriminative LDM Fitting

Require: $\mathscr{I}$, $\{\mathscr{U}_1, \ldots, \mathscr{U}_{N_i}\}$ and $\mathbf{p}$
1: **for** $i = 1$ to N_i **do**
2: $\mathbf{f} = \mathscr{F}(\mathscr{I}; \mathbf{p})$ {Get feature vector}
3: $\Delta\mathbf{p} = \mathscr{U}_i(\mathbf{f})$ {Calculate updates}
4: $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$ {Update current parameters}
5: Constrain $\mathbf{p}$.
6: **end for**
7: **return** $\mathbf{p}$

With this framework, discriminative learning then proceeds by simultaneously optimising the N_i update models, given the training set in Equation (4.1), to minimise a cost of the form:

$$\mathscr{C}(\mathscr{U}_1, \ldots, \mathscr{U}_{N_i}) = \sum_{j=1}^{N_d} \mathscr{C}_D \left(\mathscr{I}_j, \mathbf{p}_j, \mathbf{s}_j^*; \mathscr{U}_1, \ldots, \mathscr{U}_{N_i}\right), \tag{4.7}$$

where $\mathbf{s}_j^*$ are the manual annotations for the j^{th} training sample:

$$\mathbf{p}^* = \min_{\mathbf{p}} \|\mathbf{s}^* - \mathscr{S}(\mathbf{p})\|^2. \tag{4.8}$$

The distance function $\mathscr{C}_D$ in Equation (4.7) penalises the difference between the manually annotated shapes and the predicted model's shape after N_i iterations. A common choice for this is the least squares fit:

$$\mathscr{C}_D(\mathscr{I}, \mathbf{p}, \mathbf{s}^*; \mathscr{U}_1, \ldots, \mathscr{U}_{N_i}) = \left\| \mathscr{S}\left(\mathbf{p} + \sum_{i=1}^{N_i} \Delta\mathbf{p}_i\right) - \mathbf{s}^* \right\|^2. \tag{4.9}$$

Compared to texture based error measures, commonly used in generative LDM fitting, this distance function better encompasses all available information about the optimal parameter settings, i.e. the manual annotations. With this formulation, the training process essentially simulates real fitting problems on the set of training images and perturbations. If, at deployment the unseen images and their perturbations resemble those in the training set, then the fitting performance of the minimiser of Equation (4.7) can be expected to approach that at training.

Having trained the update models that minimise Equation (4.7), LDM fitting then proceeds as outlined in Algorithm 4. Notice the similarities between this method and typical fixed-update generative fitting approaches, where the main difference is that no checks need to be made regarding the reduction of texture error or the magnitude of the parameter updates to deduce convergence. Fitting is simply performed for all trained iterations with no early termination.

4.2.1 Training Complexities

Compared to the training procedure of the methods discussed in Section 2.3.2, finding the optimal update models by minimising Equation (4.7) presents a number of difficulties. Firstly, the cost function is inherently nonlinear with many local minima, due to the composition of the updates with the feature vectors and the nonlinear relationship between the pixel intensities and the warping parameters. Secondly, standard numerical optimisation techniques are not computationally practical for this problem. Since the parameter updates at each iteration depend on the update models for all previous iterations, the analytic gradient of Equation (4.7) is generally extremely complex, resulting in an impractical computational burden. This matter is made worse by the potentially large number of training samples required to ensure good generalisability of the update models.

To see this, consider a simple gradient descent on the cost function in Equation (4.7), in which the parameters of the update models at the t^{th} optimisation step takes the following form:

$$\mathbf{q}_{t+1} = \mathbf{q}_t - \eta_t \sum_{i=1}^{N_d} \frac{\partial \mathscr{C}_{D_i}}{\partial \mathbf{q}}, \tag{4.10}$$

where $\mathbf{q}$ is a concatenation of all parameters describing all N_i update models and η_t is the step size. Although gradient descent exhibits only linear convergence rates, when the derivatives can be efficiently evaluated, this approach is an attractive one due to its simplicity. However, the deterministic gradient of Equation (4.7) cannot be trivially evaluated due to the dependence of the updates on those of previous iterations. To see this, note that the deterministic gradient of $\mathscr{C}_D$ is given by:

$$\frac{\partial \mathscr{C}_D}{\partial \mathbf{q}} = \frac{\partial \mathscr{C}_D}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial \mathbf{q}} \tag{4.11}$$

where $\mathbf{z} = [\Delta\mathbf{p}_1; \ldots; \Delta\mathbf{p}_{N_i}]$ is the concatenation of the parameter updates of all iterations. The derivative $\frac{\partial \mathscr{C}_D}{\partial \mathbf{z}}$ can be easily computed from Equations (4.9) and the form of the shape generating function $\mathscr{S}$. Now, let us consider the simplest case, where the feature extractor obtains only the raw warped image:

$$\mathscr{F}(\mathscr{I}; \mathbf{p}) = \mathscr{I} \circ \mathscr{W}(\mathbf{p}). \tag{4.12}$$

and the update models take the simple linear form:

$$\mathscr{U}_i(\mathbf{f}) = \mathbf{G}_i \mathbf{f} + \mathbf{b}_i, \tag{4.13}$$

where the variables of the optimisation procedure are given by:

$$\mathbf{q} = \left[\text{vec}(\mathbf{G}_1); \ldots; \text{vec}(\mathbf{G}_{N_i}); \mathbf{b}_1; \ldots; \mathbf{b}_{N_i}\right]. \tag{4.14}$$

With this, the derivative of the j^{th} parameter update with respect to the bias vector of the k^{th}

iteration is given by the following recursive form:

$$\frac{\partial \Delta \mathbf{p}_j}{\partial \mathbf{b}_k} = \begin{cases} \mathbf{0} & \text{if } j < k, \\ \mathbf{I} & \text{if } j = k, \\ \mathbf{G}_j \left(\nabla \mathscr{I} \frac{\partial \mathscr{W}}{\partial \mathbf{p}} \right) \left(\sum_{i=1}^{j-1} \frac{\partial \Delta \mathbf{p}_i}{\partial \mathbf{b}_k} \right) & \text{if } j > k, \end{cases} \tag{4.15}$$

where the derivative of the warped image $\nabla \mathscr{I} \frac{\partial \mathscr{W}}{\partial \mathbf{p}}$ is evaluated at $\mathbf{p} + \sum_{i=1}^{j-1} \Delta \mathbf{p}_i$. The derivative with respect to the k^{th} gain matrix is given by:

$$\frac{\partial \Delta \mathbf{p}_j}{\partial \mathbf{G}_k} = \frac{\partial \Delta \mathbf{p}_j}{\partial \mathbf{b}_k} \otimes \mathscr{I} \circ \mathscr{W} \left(\mathbf{p} + \sum_{i=1}^{k-1} \Delta \mathbf{p}_i \right). \tag{4.16}$$

The evaluation of these partial derivatives is computationally intensive and memory demanding, especially those with respect to the gain matrix. Furthermore, the complexity of evaluating these partials grows exponentially with the number of iterations N_i. This problem is amplified with the use of more sophisticated feature vectors, especially those that involve a normalisation, for example that used in [30].

Therefore, optimising the discriminative learning objective simultaneously with respect to all update models is, if not intractable, very slow for most interesting problems, even for the extremely simple gradient descent minimiser.

4.2.2 Error Bound Minimisation

Although optimisation of the objective in Equation (4.7), simultaneously with respect to all update models, is not practical in general, this is not the case for a greedy learning approach, where each update model is learnt sequentially, starting with the first one to be applied to the model. As no functional compositions are involved in the optimisation, no gradients with respect to the image or warping function are required. This procedure is more akin to traditional discriminative learning where a direct mapping between the feature vector and the desired targets is learnt. However, a straightforward adaptation of matching pursuit type methods (see [79; 134], for example) for this purpose may not produce the desired outcome. The problem stems from the typical least squares criterion used. In order to accommodate the reduction of quadratic error over the whole sample set, some of the more difficult samples (i.e. those far from their optimal parameter settings) will be poorly predicted, where in some cases, they may even be perturbed further away from their desired settings. The distribution of samples in the next iteration, then, will induce a regressor that favours minimising the errors on samples that are far from their optimum at the expense of the better predicted samples. As such, the effect of using a quadratic penaliser is a cyclic pattern in the distribution of samples. No continuity between the iterations is enforced here, and convergence may be difficult to attain.

Due to the aforementioned difficulties with a matching pursuit type approach, a different objective to be greedily optimised at each iteration needs to be utilised. For this, consider first that one of the main justifications for the iterative-discriminative method is to allow simple regressors to be utilised at every iteration. As such, the performance of this method is limited by

the estimation capacity of the regressor in the final iteration. As such, when learnt simultaneously, as described in Section 4.2.1, all regressors except the last one to be applied, essentially act as *sample redistributors* in such a way that the distribution of the samples around their optimum in the last iteration can be well regressed. Therefore, the objective in greedy learning should be designed to mimic the results of a simultaneous optimisation regime.

With this in mind, consider the results by Cootes *et al.* in [30], that the relationship between the appearance residuals and the parameter updates in AAM fitting is close to linear only within a small region around the optimum of each parameter. In fact, this relationship has been shown to persist, though to a lesser extent, even for the simple warped texture feature [29]. This region is characterised by error bounds around the optimal parameter settings, within which the assumption of linearity holds relatively well. As such, if the regressors are trained in such a way that the distribution of the LDM parameters at the last iteration lie within a small error bound of the optimum for every training sample, then the use of a simple regressor (the limiting case being the linear model) in the last iteration can achieve highly accurate predictions.

The question then, is how to design an objective for the greedy training procedure, in order to achieve small error bounds on all parameters in the last iteration. It is here proposed that this can be achieved by learning a function that reduces the *error bound* in each parameter over the training set at every iteration, rather than the error itself. The idea is that although the reduction in error bound that can be afforded by simple regressors at each iteration may be small, the objective of each regressor does not need to be satisfied to a high degree, since, by virtue of the compositional regressor framework, the next regressor down the compositional line improves the global objective further, utilising new observations of the image in order to do so. Furthermore, since the error bound is reduced throughout the iterations, the distribution of the observations becomes more constrained, leading to simpler input-to-output mappings that must be estimated by the regressors. Finally, the aim of error bound reduction enforces continuity between successive iterations by virtue of their effects on the distribution of samples that they perturb. An illustration of this process is shown in Figure 4.1.

Let us denote by $\mathscr{U}^i$ the regressor for the i^{th} LDM parameter at a particular iteration. Then, the objective of iterative error bound minimisation to be minimised takes the following form:

$$\mathscr{C}_{EB} = \max \left| \mathscr{U}^i \circ \mathscr{F}\left(\mathscr{I}_j; \mathbf{p}_j^i\right) - \Delta \mathbf{p}_j^i \right| + \lambda \mathscr{R}\left(\mathscr{U}^i\right), \tag{4.17}$$

with $\mathbf{p}_j^i$ and $\Delta \mathbf{p}_j^i$ denoting the LDM parameters and their desired updates for the i^{th} parameter of the j^{th} sample, and $\mathscr{R}$ applies regularisation over the regressor to penalise over complex decision function, which is weighted by a design parameter λ. This problem can be more easily solved when posed as a constrained optimisation problem as follows:

$$\min \ \epsilon^i + \lambda \mathscr{R}\left(\mathscr{U}^i\right) \quad \text{subj to} \quad \begin{cases} \Delta p_j^i - \mathscr{U}^i \circ \mathscr{F}\left(\mathscr{I}_j; \mathbf{p}_j^i\right) \leq \epsilon \\ \mathscr{U}^i \circ \mathscr{F}\left(\mathscr{I}_j; \mathbf{p}_j^i\right) - \Delta p_i^j \leq \epsilon \\ \epsilon \geq 0 \end{cases}, \tag{4.18}$$

where ϵ is the error bound that is to be minimised. This cost function directly trades off the penalty of large error bounds over each parameter's distribution against the complexity of the update model.

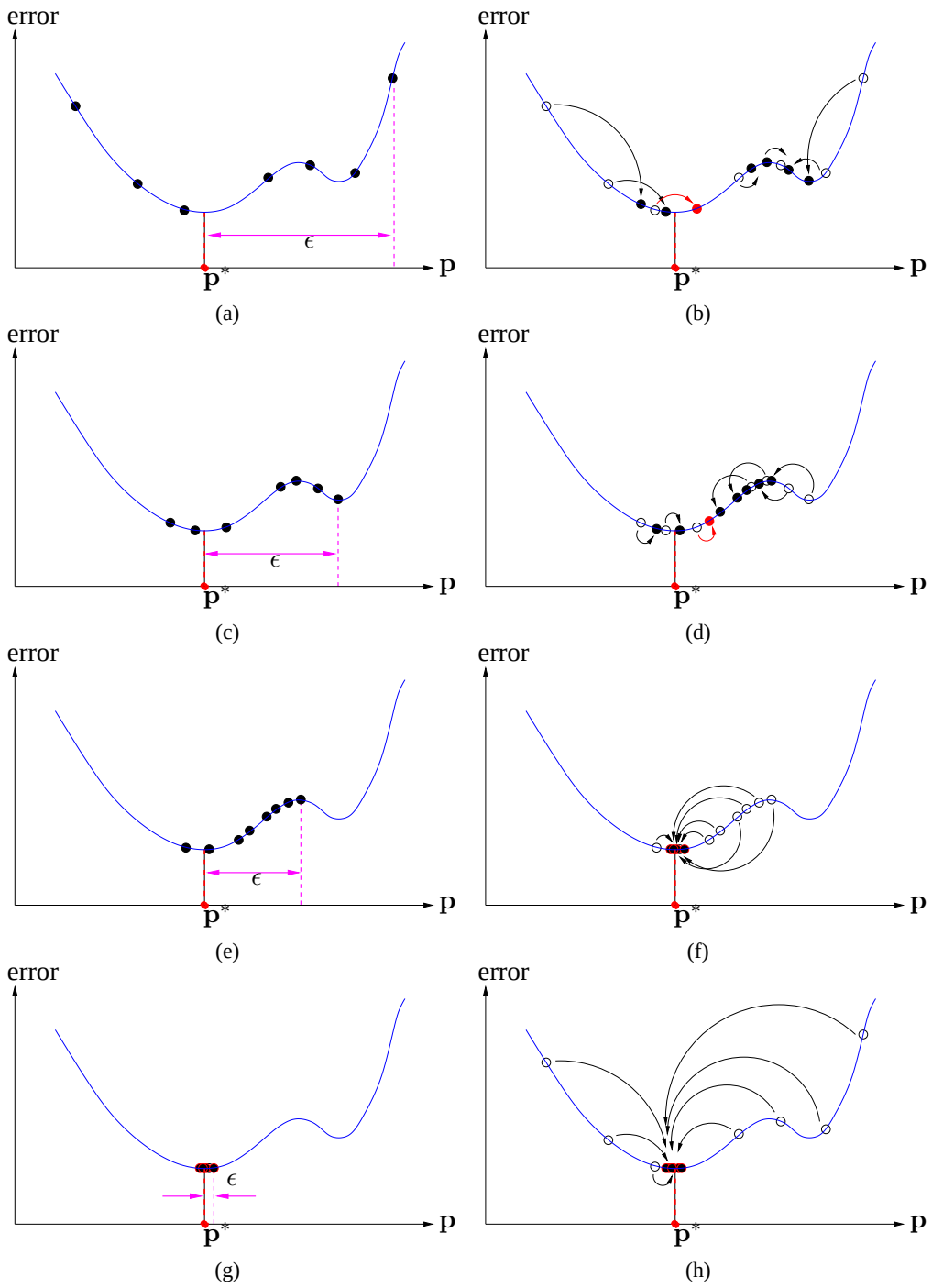


Figure 4.1: Illustration of the error bound minimisation process. Shown are perturbed samples from a *single* image, plotted on an artificial generative objective. Starting at Plot (a), a simple regressor predicts updates for all samples, as illustrated in Plot (b), yielding a new sample set with a reduced error bound, as illustrated in Plot (c). This process is continued until the desired error bound is achieved or a reduction of error bound is no longer possible (i.e. the capacity of the weak learner has been exhausted). Note that at each step it is expected that some samples will be moved *away* from the optimum (denoted by the red samples). Shown also, in Plot (h), is an illustration of the one-step discriminative fitting.

Algorithm 5 Iterative Error Bound Minimisation

Require: $\mathscr{F}$, N_i and N_d

for $i = 1$ to N_i **do**
  $\{\mathscr{I}_j, \mathbf{p}_j, \Delta\mathbf{p}_j^*\}_{j=1}^{N_d}$ {Sample training data}
  for $j = 1$ to N_d **do**
    for $k = 1$ to $i - 1$ **do**
      $\mathbf{f}_j = \mathscr{F}(\mathscr{I}_j, \mathbf{p}_j)$ {Get feature vector}
      $\mathbf{p}_j \leftarrow \mathbf{p}_j + \mathscr{U}_k(\mathbf{f}_j)$ {Update parameters}
    end for
    $\mathbf{f}_j = \mathscr{F}(\mathscr{I}_j, \mathbf{p}_j)$ {Get feature vector}
  end for
  for $j = 1$ to N_p **do**
    $\mathscr{U}_i^j \leftarrow 0$ {Initialise i^{th} update model for k^{th} parameter}
    $\mathscr{U}_i^j \leftarrow \min_{\mathscr{U}} \mathscr{E}_{EB}\left(\{\mathbf{f}_k, \Delta\mathbf{p}_k^*\}_{i=1}^{N_d}; \mathscr{U}\right)$ {Equation (4.17)}
  end for
end for
return $\mathscr{U}_1, \ldots, \mathscr{U}_{N_i}$

Although penalising the complexity of the update model encourages better generalisability, the best way to encourage generalisability is to utilise as much training data as possible in order to cover more of the input space of the system and prevent the need for functional extrapolation for inputs that are far from the those in the training set. Unlike many discriminative learning problems, LDM fitting has the peculiarity that, since the training data consists of pairs of parameter perturbations and their updates, which can be generated synthetically, for a given distribution of initial perturbation errors, the training set is potentially unbounded in size[1]. Increasing the training set size also increases the computational complexity of the training procedure. However, by virtue of the compositional form of the estimation framework in Equation (4.6), the training set size can be artificially increased without increasing the computational load in learning the regressors. At each iteration, once the optimal update model, which maximally reduces the error bounds, has been learnt for a given training set, a new set of artificial perturbations can be resampled from the initialisation distribution and propagated through all previously learnt update models in a sequential manner. The data then serves as the training set for the regressor of the next iteration. This resampling process further regularises the solution, as unseen samples that were poorly learnt previously, due to overlearning on the limited training set, are corrected. With this resampling procedure, the whole training procedure for iterative error bound minimisation is presented in Algorithm 5.

4.2.3 Variations on a Theme

Although the cost function in Equation (4.18) fulfils the objective of error bound reduction, it is not the only formulation that can achieve this. In some cases, it may be beneficial to consider

[1]Note that increases in training set size can only be accommodated in the space of deformations, not object's appearance, since the number of training images containing the object is finite.

variations on this theme, where other peculiarities about LDM fitting are incorporated into the designed cost function.

One such variation is a soft error bound minimiser. In some instances, it may be beneficial to minimise the error bound only over a subset of the training data. This may be the case when there are a few training instances that are uncharacteristically *difficult*, where the affordability of the simple regressor may be too small to be useful if it needs to accommodate these cases. These samples, for example, may be outliers in the data. To allow for this, slack variables can be utilised to capture the outliers as follows:

$$\min \;\; \lambda\mathscr{R}(\mathscr{U})+\nu\,\epsilon+\frac{1}{N}\sum_{i=1}^{N_d}(\xi_i+\hat{\xi}_i) \quad \text{subj to} \quad \begin{cases} \Delta p_i - \mathscr{U}\circ\mathscr{F}(\mathscr{I}_j;\mathbf{p}_j) \leq \epsilon+\xi_i \\ \mathscr{U}\circ\mathscr{F}(\mathscr{I}_j;\mathbf{p}_j) - \Delta p_i \leq \epsilon+\hat{\xi}_i \\ \epsilon,\xi_i,\hat{\xi}_i \geq 0 \end{cases}, \tag{4.19}$$

where the parameter index has been dropped for clarity, ξ and $\hat{\xi}$ are the slack variables and ν is a positive hyper-parameter that regulates the trade off between error bound minimisation and the influence of the outliers.

The reduction of error bounds over the data can also be utilised through the use of an asymptotic penaliser of the form:

$$\mathscr{C}_{EB} = \lambda\mathscr{R}(\mathscr{U}) + \sum_{i=1}^{N_d}\left(\epsilon - [\mathscr{U}\circ\mathscr{F}(\mathscr{I}_i;\mathbf{p}_i) - \Delta p_i]^2\right)^{-1}, \tag{4.20}$$

where

$$\epsilon \;=\; \max(\Delta p_i^2)+\delta \quad ; \quad i\in\{1,\ldots,N_d\}\;,\; \delta\in\Re^+. \tag{4.21}$$

The data term in Equation (4.20) asymptotically penalises the distance of each sample from its optimum, placing more emphasis on samples with large perturbations compared to, for example, the quadratic loss (see Figure 4.2). As such, it has the same effect of reducing the error bound, albeit indirectly. However, unlike the formulation in Equation (4.19), this cost function also penalises samples close to the optimum. Although their contribution to the total cost function is small, the asymptotic cost function ensures that the samples that are already well predicted are not *dislodged* too far in order to accommodate a reduction in error for the more perturbed samples. In Equation (4.19), no penalty is applied on perturbing samples anywhere within the error bounds; as such, there may be cases where samples cluster around the margin of the error bound, making further reduction in later iterations more difficult.

4.3 Linear and Nonlinear Prototypes

As described in Section (4.2), one of the difficulties involved in utilising discriminative learning methods is how to choose a suitable class of regressors to use on a particular problem. This problem is complicated by the requirement to select the feature extraction function that best couples with the chosen regressor. When utilising the iterative-discriminative approach, this problem becomes even more difficult since the most appropriate regressor to use, and hence its coupling feature extractor, may differ between the iterations. Nonetheless, there are a few

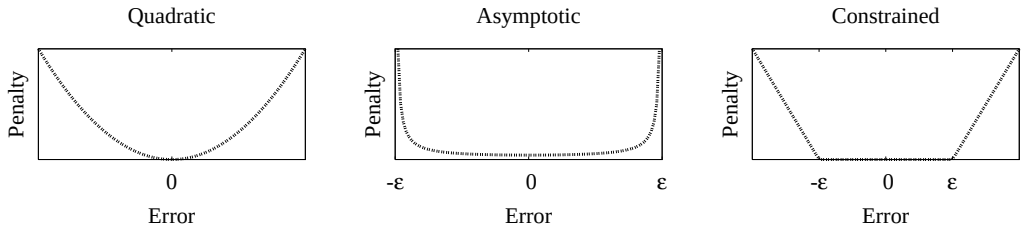


Figure 4.2: Objective functions used in iterative-discriminative fitting, along with the quadratic penaliser.

guidelines by which the choice of regressors can be steered:

- The regressor should allow for efficient evaluation to promote a rapid fitting procedure.
- The regressor should exhibit enough capacity to significantly reduce the error bound at each iteration.
- The form of the regressors should allow the optimisation of the training objective to attain a good, if not global, solution.

For most regressors, the first two guidelines can be contradictory since simple regressors generally allow efficient evaluation but exhibit poor capacity and *vice versa*.

In this section, the suitability of two classes of regressors is discussed: the linear and non-linear regressors. Details regarding their construction and training process for both variants of the iterative error bound minimisation procedure are described in detail.

4.3.1 Linear Updates

Linear regressors are by far the simplest and most efficient of regressors. They take the form:

$$\mathscr{U}(\mathbf{f}) = \mathbf{G}\mathbf{f} + \mathbf{b}, \tag{4.22}$$

where $\mathbf{G}^{(N_p \times N_f)}$ is the gain matrix , $\mathbf{b}^{(N_p)}$ is the bias and $\mathbf{f}^{(N_f)}$ is a feature vector. This update model has been used successfully in a number of LDM fitting methods, most notably in the AAM literature [43; 83]. However, the utility of this model in most generative fitting methods is limited by two factors:

- A fixed linear update model cannot accurately account for the various error terrains about the optimum in different images. The linear regression [43] and project-out [83] methods, for example, exhibit this drawback.
- Utilising adaptive linear update models usually requires a costly process of re-calculating it for every iteration. The adaptive [12] and the simultaneous inverse compositional [4] methods, for example, exhibit this drawback.

However, it is argued here that when utilising a fixed update model within the compositional framework of Equation (4.6), coupled with a training regime that reduces the error bounds over

the data, the full capacity of this simplest of regressors can be taken advantage of. Due to the simplicity of linear models, they are expected to only reduce the error bounds marginally at every iteration. However, when combined, the total reduction in error bound may be sufficient for many applications.

Details regarding the training procedures for both the constrained optimisation and asymptotically penalised objective are presented below. Since regressors for each parameter are trained separately, in the following discussions let:

$$\mathbf{G} = \left[\mathbf{g}_1^T; \ldots; \mathbf{g}_{N_p}^T\right] \quad \text{and} \quad \mathbf{b} = \left[b_1; \ldots; b_{N_p}\right], \tag{4.23}$$

where the subscript denoting parameter indices has been dropped for clarity of exposition. With this, the regressor for any parameter takes the form:

$$\mathscr{U}(\mathbf{f}) = \langle \mathbf{g}, \mathbf{f} \rangle + b. \tag{4.24}$$

The regularisation used for this linear model is performed separately for each parameter, and takes the form of an L_2-norm of the gain vector:

$$\mathscr{R}(\mathscr{U}) = \mathbf{g}^T\mathbf{g}. \tag{4.25}$$

Asymptotically Penalised Training

Utilising a linear regressor in the error bound reduction objective in Equation (4.20), the problem now takes the form:

$$\mathscr{C}_{EB} = \lambda \mathbf{g}^T\mathbf{g} + \sum_{i=1}^{N_d} \left(\epsilon - [\langle \mathbf{g}, \mathbf{f}_i \rangle + b - \Delta p_i]^2\right)^{-1}. \tag{4.26}$$

Note that this asymptotic penaliser is convex within the convex set:

$$\left\{(\mathbf{g}, b) \;\middle|\; -\sqrt{\epsilon} < \langle \mathbf{g}, \mathbf{f}_j \rangle + b - \Delta p_j < \sqrt{\epsilon} \;\; , j = 1, \ldots, N_d\right\}, \tag{4.27}$$

i.e. the intersection of N_d convex sets, each composed of the region between two parallel hyperplanes. Due to the choice of ϵ in Equation (4.21), the null model ($\mathscr{U} \leftarrow \mathbf{0}$) lies within this convex region. As such, starting with the null model and performing steepest descent with a line search allows the globally optimum update model to be found.

Unlike the optimisation problem discussed in Section 4.2.1, the gradient of this cost function is easily computed:

$$\frac{\partial \mathscr{C}_{\text{EB}}}{\partial b} = \sum_{i=1}^{N_d} \theta_i \quad \text{and} \quad \frac{\partial \mathscr{C}_{\text{EB}}}{\partial \mathbf{g}} = 2\lambda\mathbf{g} + \sum_{i=1}^{N_d} \theta_i \mathbf{f}_i, \tag{4.28}$$

where:

$$\theta_i = 2r_i\left(\epsilon - r_i^2\right)^{-2} \quad ; \quad r_i = \langle \mathbf{g}, \mathbf{f}_i \rangle + b - \Delta p_i. \tag{4.29}$$

Although second order optimisation methods, such as the Newton method, are not generally

viable for this problem, due to the high dimensionality of $\mathbf{g}$, efficient first order methods can be utilised here. One example is the limited memory BFGS algorithm (L-BFGS) [74], a variant of the quasi-Newton optimiser BFGS, which avoids the cost of storing and updating the estimate of the cost function's Hessian inverse. Given the L-BFGS step direction $\mathbf{d}$, the line search is performed by solving:

$$\alpha^* = \min_{\alpha} \;\; \alpha\lambda\|\mathbf{d}\|^2 + \sum_{j=1}^{N_d} \Big(\epsilon - [\alpha\langle \mathbf{d}, [\mathbf{f}_j; 1]\rangle + \langle \mathbf{g}, \mathbf{f}_j\rangle + b - \Delta p_j]^2\Big)^{-1} \tag{4.30}$$

subject to:

$$0 \leq \alpha \leq \min\left(\frac{\pm\sqrt{\epsilon} - \langle \mathbf{g}, \mathbf{f}_j\rangle - b + \Delta p_j}{\langle [\mathbf{f}_j; 1], \mathbf{d}\rangle}\right) \;\; ; \;\; j \in \{1, \ldots, N_d\}, \tag{4.31}$$

where the sign of $\sqrt{\epsilon}$ is chosen to represent the asymptote in the direction of the update. In fact, due to the convexity of the cost function within the bounds on α, convex line search methods [38] can be utilised to achieve rapid optimisation.

Constrained Optimisation Training

Utilising a linear regressor in the error bound reduction objective in Equation (4.18), the problem becomes that of a quadratic program:

$$\min \;\; \frac{\lambda}{2}\mathbf{g}^T\mathbf{g} \;+\; \epsilon \quad \text{subj to} \quad \begin{cases} \Delta p_i - \langle \mathbf{g}, \mathbf{f}_i\rangle - b \leq \epsilon \\ \langle \mathbf{g}, \mathbf{f}_i\rangle + b - \Delta p_i \leq \epsilon \\ \epsilon \geq 0 \end{cases} . \tag{4.32}$$

This formulation can generate a globally optimal solution for the parameters $\mathbf{g}$ and b. However, more interesting perhaps is the formulation obtained from Equation (4.19), which yields:

$$\min \;\; \frac{\lambda}{2}\mathbf{g}^T\mathbf{g} + \nu\,\epsilon + \frac{1}{N}\sum_{i=1}^{N}(\xi_i + \hat{\xi}_i) \quad \text{subj to} \quad \begin{cases} \Delta p_i - \langle \mathbf{g}, \mathbf{f}_i\rangle - b \leq \epsilon + \xi_i \\ \langle \mathbf{g}, \mathbf{f}_i\rangle + b - \Delta p_i \leq \epsilon + \hat{\xi}_i \\ \epsilon, \xi_i, \hat{\xi}_i \geq 0 \end{cases} . \tag{4.33}$$

The resulting problem then becomes that of the linear ν-Support Vector Regression (ν-SVR) method [108]. As the training of support vector regression is a global optimisation process, a global solution for the gain and bias is also obtained. Therefore, one of the advantages of utilising a linear model within this framework is that off-the-shelf ν-SVR learners can be directly utilised for training here, where rapid training procedures, such as sequential-minimal optimisation [92], have been implemented.

Based on work on ν-SVR, further insight into the role of the hyper-parameter ν can be gained. Of interest to the problem of LDM fitting, it denotes the upper bound on the fraction of samples that lie outside the error bound ϵ. This can be very useful when an estimate of the number of outliers or difficult cases is known *a-priori*. In turn, this implies that choosing $\nu \geq 1$

will result in $\epsilon = 0$ that is equivalent to minimising the L-1 norm over the whole training set with respect to the model parameters. In general, however, ν should be chosen to be a small fraction, such as 0.001, in order to enforce the objective of error bound minimisation.

4.3.2 Nonlinear Updates

Due to the limited predictive capacity of linear models, it may be necessary for some problems to utilise a nonlinear regressor to obtain accurate results. The main difficulty in utilising nonlinear regressors is choosing an appropriate one for the task. Since there is a plethora of nonlinear regression forms to choose from, this problem is non-trivial in general.

Nonetheless, following the guidelines defined in Section (4.3), in this section, two suitable regressors for the problem of LDM fitting are proposed. In each case, regressors for each parameter in each iteration are trained separately from each other. The final update model, then, is a concatenation of the updates for every parameter:

$$\Delta\mathbf{p} = \mathscr{U} \circ \mathscr{F}(\mathscr{I}; \mathbf{p}) = \left[\mathscr{U}^1(\mathbf{f}); \ldots; \mathscr{U}^{N_p}(\mathbf{f})\right], \tag{4.34}$$

As with the linear case, simultaneous training of the regressors for each parameter requires prior information regarding the relationships between the perturbations of each parameter, which is not generally available. Furthermore, this prior information may be difficult to integrate within the error bound minimisation framework. It should be noted however, that the relationship between the perturbations is implicitly encoded into the procedure through the compositional framework, where the training set for a particular iteration is a result of predictions in previous iterations over all parameters.

In this thesis, a linear expansion of weak learners is proposed as the prototype of nonlinear regressor to use. Formally, the update function for the k^{th} parameter at any iteration takes the following form:

$$\mathscr{U}^k(\mathbf{f}) = \sum_{t=1}^{N_b} \alpha_t^k \mathscr{L}_t^k(\mathbf{f}) \;\; ; \;\; \mathscr{L}_t^k \in \mathcal{L}, \tag{4.35}$$

where $\mathscr{L}_t^k$ is a weak nonlinear learner, a number of which can combine to form a strong ensemble $\mathscr{U}^k$. Here, $\mathcal{L}$ is a *dictionary* of weak learners. The choice of regressor within this prototype then depends on the scope of $\mathcal{L}$.

Asymptotically Penalised Training

There are two requirements of the weak function set $\mathcal{L}$ for discriminative fitting. Firstly, their evaluation must be computationally cheap, such that efficient fitting can be achieved with a reasonably sized ensemble. Secondly, they must be sufficiently rich, such that complex regression functions can be accurately estimated by a linear combination of them. The Haar-like feature set $\mathcal{H}$, popularised by Viola and Jones in [135], acts as a good basis for the weak function set as they fulfil both of the required criteria: efficient evaluation using the integral image and a capacity for complex representations through their similarity to Haar wavelets. In particular, extensions to the original Haar-like features [72] may also be beneficial to consider, as they include diagonal features, useful for approximating rotations.

For the asymptotically penalised error-bound objective utilising this type of weak learner, a boosting-like procedure appears to be a viable solution. Starting with an empty ensemble, one weak learner at a time is appended to the ensemble:

$$\mathscr{U}_{t+1}^{k} = \mathscr{U}_{t}^{k} + \alpha_{t}^{k}\mathscr{L}_{t}^{k}, \tag{4.36}$$

choosing $(\alpha_t^k, \mathscr{L}_t^k)$ to maximally decrease the objective function for each addition. Utilising this boosted regressor in the error bound reduction objective in Equation (4.20), the problem now takes the form:

$$\mathscr{C}_{EB}(\alpha_T, \mathscr{L}_T) = \sum_{i=1}^{N_d} \left(\epsilon - \left[\Delta p_i - \sum_{t=1}^{T} \alpha_t \mathscr{L}_t(\mathbf{f}_i) \right]^2 \right)^{-1}, \tag{4.37}$$

for the T^{th} learner to be added to an ensemble at a given iteration, subject to $\alpha_T \in [a, b]$ and $\mathscr{L}_T \in \mathcal{L}$, where the parameter index k has been dropped for clarity. Here,

$$a = \max\left(\frac{(\Delta\hat{p}_i)^2 - \epsilon}{\mathscr{L}_T(\mathbf{f}_i)} \right) \;\;,\;\; b = \min\left(\frac{(\Delta\hat{p}_i)^2 + \epsilon}{\mathscr{L}_T(\mathbf{f}_i)} \right) \tag{4.38}$$

with:

$$\Delta\hat{p}_i = \Delta p_i - \sum_{t=1}^{T-1} \alpha_t \mathscr{L}_t(\mathbf{f}_i) \tag{4.39}$$

being the current residual target updates after $T-1$ learners have been added to the ensemble. As each entry in the sum is convex, the objective of each round of boosting is also convex. Therefore, for a given $\mathscr{L}_T(\mathbf{f})$, the optimal α_T can be found through a 1D line search between a and b. Again, note that since the cost is convex, convex line search procedures can be utilised here, to rapidly find the best solution.

To regularise the solution, shrinkage is performed on the ensemble [48]. This common regularising method involves shrinking the optimal α for the newly selected $\mathscr{L}$ by a factor $\eta \in [0, 1]$ before adding it to the ensemble. This approach is preferable compared to a direct regularisation term in the cost function as in Equation (4.20) since the weak learners are added one at a time.

A common choice of $\mathcal{L}$ that utilises these features is the one-dimensional decision stump:

$$\mathscr{L}(\mathbf{f}) = \begin{cases} +1 & \text{if } s\mathscr{H}(\mathbf{f}) \geq s\theta \\ -1 & \text{otherwise} \end{cases}, \tag{4.40}$$

where $\mathscr{H}$ is a Haar-like filtering function, θ is a decision threshold and $s \in \{1, -1\}$ is a parity direction indicator. Although this weak function has been utilised in many works, for example [72; 135; 147], it has some major drawbacks. Firstly, most functions in this set are non-discriminative in the sense that, for a given $\mathscr{H}$, the best choice of s and θ will still result in a poor $\mathscr{L}$. Secondly, for those which are discriminative, the optimal choice of s and θ must be found through trial and error, an expensive process. This is especially potent in a discriminative fitting problem, where the size of $\mathcal{H}$ is extremely large due to the size of the image region to

Algorithm 6 Weak Learner Sampling Algorithm

Require: $\{\mathbf{f}, \Delta\mathbf{p}\}_{j=1}^{N_d}$, $\mathcal{H}$ and n_b

1: Calculate weight of each sample: $(\epsilon - |\Delta p_i|)^{-1}$
2: Sample a Haar-like feature $\mathscr{H} \in \mathcal{H}$ {see [72]}
3: Build H_+ and H_- histograms {Eqn. (4.42) & (4.43)}
4: Compute weak learner $\mathscr{L}$ {Eqn. (4.41)}
5: Find optimal α through 1D line search {Eqn. (4.37)}
6: **return** $(\alpha, \mathscr{L})$

be analysed, which is around 5 to 10 times that of the images used in [135; 147].

Rather than using the weak function set described above, the response binning approach in [96], which maximises the utility of weak learners derived from the Haar-like features, will be followed here. In their method, the weak learners of a classification problem, take the form of the relative inequality between histograms of the positive and negative examples:

$$\mathscr{L}(\mathbf{f}) = \begin{cases} +1 & \text{if } H_+(\mathscr{H}(\mathbf{f})) > H_-(\mathscr{H}(\mathbf{f})) \\ -1 & \text{otherwise} \end{cases} \tag{4.41}$$

where H_+ and H_- are 1D histograms of the distribution of the feature evaluations on the positive and negative examples, respectively. This method affords a multimodal decision surface whilst maintaining efficiency, as it requires only a table lookup for its evaluation.

To adapt this approach for regression, a few modifications need to be made. The objective function to be minimised in Equation (4.37) aims to reduce the spread of the training data about the optimum. Therefore, in formulating $\mathscr{L}$, preference should be made on reducing the error over samples with large, compared to small, error. To this end, $H_{+/-}$ is defined as the histogram of weighted samples with positive/negative target values:

$$H_+(v) = \sum_{\mathscr{H}(\mathbf{f}_i)=v} \frac{1}{\epsilon - \Delta\hat{p}_i} \;\; ; \;\; \Delta\hat{p}_i > 0 \tag{4.42}$$

$$H_-(v) = \sum_{\mathscr{H}(\mathbf{f}_i)=v} \frac{1}{\epsilon + \Delta\hat{p}_i} \;\; ; \;\; \Delta\hat{p}_i < 0, \tag{4.43}$$

where $\Delta\hat{p}_i$ is given in Equation (4.39). The idea here to build $\mathscr{L}$, such that the functional direction is in that which reduces the error over the most difficult samples in each bin, with the aim of reducing sample spread.

The only parameter that needs adjusting for this weak function set is the number of bins in the histograms n_b. Too many bins may cause overlearning in sparsely sampled bins, but too few bins may not capture enough of the nonlinearity of the target function, limiting the capacity of these learners. In this thesis, n_b is fixed at an empirically good value and overlearning is avoided by setting $\mathscr{L}$ at sparsely sampled bins to zero (i.e. avoid making decisions that are not well supported by the training data). A summary of the generation of a weak learner is given in Algorithm 6.

Constrained Optimisation Training

Noting the reduction of the constrained problem in Equation (4.18) to a linear ν-SVR problem in Section 4.3.1, a straightforward extension to a nonlinear regressor can be obtained by projecting the feature vector into a reproducing kernel Hilbert space leading to the nonlinear ν-SVR formulation [108]. This procedure is viable for this problem since the algorithm can be cast solely in terms of dot products in Hilbert space, which can be expressed through a positive definite kernel:

$$\mathscr{K}(\mathbf{x}_1, \mathbf{x}_2) \colon \Re^{N_f} \times \Re^{N_f} \rightarrow \Re = \langle \psi(\mathbf{x}_1), \psi(\mathbf{x}_2) \rangle \tag{4.44}$$

where ψ is the nonlinear map that relates the input space to Hilbert space. The choice of $\mathscr{K}$ will generally be problem dependent, but typical choices include the radial basis function:

$$\mathscr{K}(\mathbf{x}_1 - \mathbf{x}_2) = \exp\left\{\frac{-\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2\sigma^2}\right\} \quad ; \quad \sigma > 0 \tag{4.45}$$

and the polynomial kernel:

$$\mathscr{K}(\mathbf{x}_1, \mathbf{x}_2) = \langle \mathbf{x}_1, \mathbf{x}_2 \rangle^d \quad ; \quad d \in N \tag{4.46}$$

Although these nonlinear regressors certainly have much greater capacity than the simple linear kernel, they are much more expensive to evaluate. In the linear case, the support vectors lie within the input space, allowing the gain vector to be obtained directly as a linear combination of the support vectors:

$$\mathbf{g} = \sum_{i}^{N_k} \beta_i \mathbf{v}_i, \tag{4.47}$$

where $\mathbf{v}_i$ is the i^{th} support vector out of N_k and β_i is related to the dual variables of the ν-SVR formulation (see [108]). For nonlinear kernel types, the support vectors live in the Hilbert space, preventing them from being evaluated explicitly, instead taking the form:

$$\mathscr{U}(\mathbf{f}) = \sum_{i=1}^{N_k} \alpha_i \mathscr{K}(\mathbf{f}, \mathbf{v}_i), \tag{4.48}$$

where α_i is the expansion coefficient for the i^{th} support vector.

The main reason for the efficiency penalty, when using nonlinear kernels, is due to the evaluation of the kernel function, which generally involves an inner product between two vectors of the size of the observation. Despite the various claims of sparsity, support vector algorithms are notorious for keeping a large number of support vectors in comparison to other methods, such as the relevance vector machine [124] or kernel matching pursuit [134]. As such, these kernel evaluations generally need to be performed over a large number of support vectors.

In order to build an efficient fitting procedure utilising the kernel expansion as a regressor, the dimensionality of the feature vector to be evaluated by the kernel function must be reduced. A number of methods exist for dimensionality reduction, however, in this thesis, PCA will be used exclusively since it allows the dimensionality of the data to be reduced by a factor of 10 in

many problems whilst sacrificing little accuracy. In fact, as opposed to modelling appearance of registered data (such as in LDM model building), the major modes of variation obtained by applying PCA to the feature vectors will stem, in a large part, from the misalignments rather than variations of appearance within the objects class. As such, PCA is a natural choice for dimensionality reduction here, since it preserves the features that are important to fitting (i.e. those that pertain to misalignment).

The typical PCA expansion is given by:

$$\mathbf{f} = \bar{\mathbf{f}} + \mathbf{\Phi}\mathbf{b}, \tag{4.49}$$

where $\bar{\mathbf{f}}$ is the mean feature vector, $\mathbf{\Phi}$ is the matrix of modes of variation and $\mathbf{b}$ is the PCA expansion coefficients. To accommodate for the distribution of appearance of the feature vectors, a separate basis matrix $\mathbf{\Phi}$ is learnt for each iteration. This way, representational power is concentrated on the problem at hand. To account for global lighting variations, the feature vectors can be normalised to a mean of zero and a variance of one before applying PCA. With this, each weak learner in the linear expansion form in Equation (4.35), takes the form:

$$\mathscr{L}_i(\mathbf{f}) = \mathscr{K}\left(\mathbf{\Phi}^T\left[\mathscr{N}(\mathbf{f}) - \bar{\mathbf{f}}\right] \;\;,\;\; \mathbf{v}_i\right), \tag{4.50}$$

where $\mathscr{N}$ is the lighting normalising function. By applying ν-SVR learning on the expansion coefficients, a rapid fitting procedure can be obtained.

4.4 Robustification

One of the major difficulties in utilising discriminative approaches for LDM fitting is how to robustify the algorithm against outliers due to occlusional effects or appearance variation not present in the training set. Since discriminative methods learn an input-to-output mapping function, if the types of occlusion are predictable, then by including examples of the occluded object in the training set, a mapping function that is robust towards these occlusions can be learnt. However, in general, the types of occlusions are not known *a-priori*. Furthermore, even if the types of occlusions are known, for most cases, an extremely large training set will be required to accommodate the various instantiations of each occlusion. This in turn will increase the computational complexity of the training procedure, for example, when occlusions occur due to objects lying in the line of sight between the object and the camera. A training set accounting for this very general type of occlusion must include examples of varying sizes, appearance, shape and location of the occluding object, an intractable task for most discriminative learning algorithms.

In generative LDM fitting, however, significant advances have been made to robustify the fitting procedures. The main reason for successful robustification here is by virtue of the generative framework, where model fit, and hence convergence, is measured through the difference between the LDM's synthesised appearance and that of the image, warped to the canonical frame using the LDM's synthesised shape. Outlying image pixels generally exhibit much larger differences with the synthesised appearance compared to inliers when the LDM is well aligned to the object in the image. By utilising a robust error function, which penalises large er-

rors less severely than the typical quadratic penaliser, an objective function can be constructed that shares the same global optimum as the true problem. Optimisation of this objective is typically performed through an iteratively reweighted least squares formulation.

There are two main difficulties with robust generative fitting, however. Firstly, the iteratively reweighted least squares formulation does not allow a fixed update model to be precomputed since it depends on the weights allocated to the various components of the cost function. A straight forward implementation, using for example the Gauss-Newton procedure, will result in inefficient fitting. This difficulty is addressed in [55] by assuming spatial coherence of outliers, assigning the average weight of a region in the canonical frame to all error components pertaining to that region. As a result, a significant proportion of the computation can be precomputed[2], allowing a real time fitting algorithm to be implemented (albeit for a simple person-specific model). More recently in [99], the authors consider contributions of specific errors to a single LDM parameter, allowing the fixed linear regression matrix to be utilised. Only the feature vectors is modified to reflect the confidence regarding the outlier likelihood of a feature. Although potentially more efficient than that in [55] for models with a large number of parameters, the predicted parameter updates are only approximations to the true iteratively reweighted least squares, biased in favour of inliers of the cost function.

The second difficulty with robust generative fitting is an artifact of the generative construction itself. When the model is misaligned, such as in the early stages of fitting, the distinction between inliers and outliers is difficult to deduce. As such, downweighting the contribution of pixels with large errors, or completely excluding them as in [104], may ignore useful information from inlier pixels that exhibit large errors due to misalignment. In fact, most information regarding misalignment is contained within pixels with large error. For this reason, ignoring these pixels in the fitting procedure may lead to slow convergence or even convergence failure. This problem is directly addressed in [97], where the multimodal nature of the error histogram is analysed in order to distinguish between inliers and outliers. The method discards the concept of domain thresholds for inlier and outlier errors. Instead, error modes are selected for inclusion in the optimisation procedure according to their impact on the matching process. However, the mode analysis procedure is computationally expensive, leading to inefficient fitting. In [99], the authors partially address this problem by performing a form of deterministic annealing, where two sets of robust scaling parameters are chosen to account for errors in the early and later stages of fitting. However, since at early stages the scaling factor is chosen to include large errors, the estimations at these early iterations may be severely influenced by outliers, leading to significant perturbations.

4.4.1 Robust Feature Extraction

To robustify the discriminative fitting methods described in this chapter, the sources of difficulty regarding outliers must first be considered. In general, including occluded instances into the training set is not viable. However, the space of unoccluded instances is generally restricted to a smaller, albeit nonlinear, subspace. As such, an unoccluded feature vector can be represented using a parameterisation in this reduced space. For example, consider the non-

[2]Note that the spatial coherence assumption results in precomputable parts since it is implemented within an inverse-compositional framework.

linear kernel regression prototype described in Section (4.3.2), where the feature vector is represented as a linear combination of modes of variation. Extraction of the reduced subspace coordinates, used for regression through a nonlinear kernel, is performed via a least squares fit between the true and synthesised feature vector:

$$\mathscr{E}_{LS}(\mathbf{b}) = \|\mathbf{f} - \left(\bar{\mathbf{f}} - \mathbf{\Phi}\mathbf{b}\right)\|^2. \tag{4.51}$$

When encountering a feature vector with outliers, this least squares fit will be biased towards the outliers. However, a robust error norm is utilised here, as in the case of generative fitting, the effects of outliers on the estimation of the reduced space coordinates can be lessened.

Consider the robust least squares fit:

$$\mathscr{E}_{RLS}(\alpha, \beta, \mathbf{b}) = \sum_{i=1}^{N_f} \psi\left(\left[\mathbf{f}_{(i)} - \alpha\left(\bar{\mathbf{f}}_{(i)} + \mathbf{\Phi}_{(i,:)}\mathbf{b}\right) - \beta\right]^2; \sigma_i\right), \tag{4.52}$$

where σ_i is the robust scaling parameter for the i^{th} feature, α denotes the global lighting scaling and β the bias. Note that the inclusion of global lighting parameters is required here since normalising the feature vector, as described in Section (4.3.2), will include the outlier effects in the normalisation, which in turn will result in a biased estimate of $\mathbf{b}$. The aim of robust feature extraction, then, is to obtain the parameters $\mathbf{b}$ that minimise the cost function in Equation (4.52). This nonlinear problem can be solved by iteratively approximating the problem as one of weighted least squares. Using the change of variable $\mathbf{p} = [\alpha; \beta; \mathbf{b}]$, the cost function can be written:

$$\mathscr{E}_{RLS}(\mathbf{p}) = \sum_{i=1}^{N_f} \psi\left(\left[\mathbf{f}_{(i)} - \hat{\mathbf{\Phi}}_{(i,:)}\mathbf{p}\right]^2; \sigma_i\right) \quad \text{where} \quad \hat{\mathbf{\Phi}} = \begin{bmatrix}\bar{\mathbf{f}} & \mathbf{1} & \mathbf{\Phi}\end{bmatrix}. \tag{4.53}$$

Letting $\mathbf{p} = \mathbf{p}^c + \Delta\mathbf{p}$, then expanding the squared term within the robust penaliser, and taking a first order Taylor expansion of each penaliser around the current estimation error, Equation (4.53) can be approximated by:

$$\begin{aligned}\mathscr{E}_{RLS}(\mathbf{p}) \approx \sum_{i=1}^{N_f} \psi\left(\left[\mathbf{f}_{(i)} - \hat{\mathbf{\Phi}}_{(i,:)}\mathbf{p}^c\right]^2; \sigma_i\right) + \nabla\psi\left(\left[\mathbf{f}_{(i)} - \hat{\mathbf{\Phi}}_{(i,:)}\mathbf{p}^c\right]^2; \sigma_i\right) \times \\ \left[\Delta\mathbf{p}^T\hat{\mathbf{\Phi}}_{(i,:)}^T\hat{\mathbf{\Phi}}_{(i,:)}\Delta\mathbf{p} - 2\mathbf{p}^T\hat{\mathbf{\Phi}}_{(i,:)}^T\left(\mathbf{f}_{(i)} - \hat{\mathbf{\Phi}}_{(i,:)}\mathbf{p}^c\right)\right].\end{aligned} \tag{4.54}$$

Taking the derivative of this approximated form with respect to $\Delta\mathbf{p}$, and equating it with zero, the incremental updates, which are to be applied additively to $\mathbf{p}$, are then given by:

$$\Delta\mathbf{p} = \left[\sum_{i=1}^{N_f} \omega_i \hat{\mathbf{\Phi}}_{(i,:)}^T \hat{\mathbf{\Phi}}_{(i,:)}\right]^{-1} \sum_{i=1}^{N_f} \omega_i \left(\mathbf{f}_{(i)} - \hat{\mathbf{\Phi}}_{(i,:)}\mathbf{p}^c\right) \hat{\mathbf{\Phi}}_{(i,:)}^T, \tag{4.55}$$

where $\omega_i = \nabla\psi\left(\left[\mathbf{f}_{(i)} - \hat{\mathbf{\Phi}}_{(i,:)}\mathbf{p}^c\right]^2; \sigma_i\right)$ is the derivative of the i^{th} robust function, evaluated

at the current squared estimation error. The features used to regress an update are then given by:

$$\mathbf{b} = \frac{1}{\mathbf{p}^*_{(1)}}\mathbf{p}^*_{(3:)}, \tag{4.56}$$

where $\mathbf{p}^*$ is the solution to Equation (4.53).

Examining the forms of the parameter update in Equation (4.55), one notices that no derivatives with respect to the image pixels or warping functions need to be computed. As such, compared to its relating form in generative fitting, the updates here are significantly more efficient to evaluate. However, unlike the fitting problem in a generative formulation, the robust fitting must be performed until convergence for each LDM parameter update, since the subspace representation changes throughout the fitting procedure to account for the distribution of feature vectors at the various error bounded regions. As such, the fitting procedure can still be quite computationally demanding when evaluated as is.

The main bottleneck of the computations here is the computation of the Gauss-Newton Hessian that involves N_f additions of Hessian sized matrices. One way to reduce the computation here is to assume a degree of spatial coherence of the outliers as in [55]. Rather than assuming each component of the feature vector is weighted separately, a single weight is applied to components of the feature vector that exhibit spatial coherence. Subdividing the feature vector into N_c non-overlapping, spatially coherent regions:

$$\mathcal{R} = \{\mathcal{R}_1 \cup \ldots \cup \mathcal{R}_{N_c}\}, \tag{4.57}$$

the Hessian can be approximated as:

$$\sum_{i=1}^{N_f} \omega_i \hat{\mathbf{\Phi}}^T_{(i,:)} \hat{\mathbf{\Phi}}_{(i,:)} \approx \sum_{k=1}^{N_c} \varphi_k \mathbf{H}_k, \tag{4.58}$$

where:

$$\mathbf{H}_k = \sum_{i \in \mathcal{R}_k} \hat{\mathbf{\Phi}}^T_{(i,:)} \hat{\mathbf{\Phi}}_{(i,:)}, \tag{4.59}$$

and φ_k is set, for example, to the mean weights within $\mathcal{R}_k$ [55]:

$$\varphi_k = \frac{1}{\text{size}(\mathcal{R}_k)} \sum_{i \in \mathcal{R}_k} \omega_i. \tag{4.60}$$

Other possibilities for φ_k include the median or mode of the distribution [4]. With this approximation, all regional Hessians $\{\mathbf{H}_i\}_{i=1}^{N_c}$ can be precomputed, resulting in only N_c, compared to N_f, Hessian sized matrix additions to compute the Hessian.

Although the spatial coherence approximation can significantly reduce online computational costs, since the whole appearance fitting procedure must be performed once for each iteration of LDM fitting, the resulting algorithm is still comparably slow. The optimisation at each step can be accelerated significantly, however, if a good initialisation is available. Since the expansion coefficients $\mathbf{b}$ describe the appearance of the image at the current settings of the LDM's shape, and the update model $\mathscr{U}$ predicts perturbations to the LDM's shape parame-

ters, which in turn give rise to the appearance of the image at some new shape setting, there may exist a mapping between the parameters $\mathbf{b}$ and those in the next iteration of the fitting procedure:

$$\mathscr{M} : \Re^{N_{f_i}} \rightarrow \Re^{N_{f_{i+1}}}, \tag{4.61}$$

where N_{f_i} denotes the number of appearance expansion coefficients for the i^{th} iteration. Since the dimensionality of the expansion coefficients is relatively low, compared to that of the original feature vector $\mathbf{f}$, sophisticated regressors can be learnt for this mapping, including Support Vector Regression, Neural Networks or boosting type approaches. However, in this thesis, only the utility of a linear regressor for this purpose is investigated since it allows a rapid evaluation, it does not involve any free parameters to be tuned and optimal solutions can be obtained in closed form. Specifically, given a trained nonlinear iterative-discriminative fitting model, as described in Section 4.3.2, a set of optimal expansion coefficients for each iteration can be obtained by fitting the model to outlier-free images. The mapping function can then be learnt by solving the linear system:

$$\mathbf{M}_{i+1}^{(N_{f_i} \times N_{f_{i+1}})} \begin{bmatrix} \mathbf{b}_1^i & \dots & \mathbf{b}_N^i \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1^{i+1} & \dots & \mathbf{b}_N^{i+1} \end{bmatrix}, \tag{4.62}$$

with respect to $\mathbf{M}$, where N denotes the number of fitting trials performed by the non-robust fitting model. Although the linear model has a restricted predictive capacity, a highly accurate mapping function is not necessary here, since it is used only to obtain a reasonable initialisation for the optimisation of Equation (4.53).

Along with the two efficiency promoting modifications described above, a summary of the robust feature extraction algorithm is outlined in Algorithm 7. Notice that in the first iteration, the appearance is initialised to its average (i.e. $\mathbf{b} = 0$), which is the best initialisation when no other information is available. Also note that the lighting parameters (α, β) of a previous iteration are used as an initial estimate in the current iteration.

4.4.2 Independent Robust Scalings

In the formulation for robust feature extraction in Equation (4.52), one immediately notices the use of different robust scaling parameters for each element in the summation. These robust scalings should generally be set such that inliers are quadratically penalised, assuming Gaussian noise, with a decreasing rate of penalty for outliers. For example, in the Gemman-McClure robust penaliser [15; 16]:

$$\psi(r; \sigma) = \frac{r^2}{r^2 + \sigma^2}, \tag{4.63}$$

the inlier region, where errors are penalised quadratically, is $\left[-\frac{\sigma}{\sqrt{3}}, \frac{\sigma}{\sqrt{3}}\right]$, with a decreasing rate of penalty outside of it. In the case where the linear expansion in Equation (4.49) is chosen such that all modes relating to variations other than those due to noise are retained, separate robust scalings for each term are not required. This is because the remaining directions pertaining to image noise can generally be assumed to exhibit similar variance (i.e. the eigenspectrum for these modes is relatively flat).

However, as discussed in Section 4.3.2, since most information pertaining to misalignment

Algorithm 7 Robust Feature Extraction with Spatial Coherence

Require: $\mathbf{b}^{i-1}, \alpha^{i-1}, \beta^{i-1}, \mathbf{f}^i, \mathcal{R}^i, \{\mathbf{H}^i_j\}_{j=1}^{N^i_c}, \mathbf{M}^i, \hat{\mathbf{\Phi}}^i$

if $i = 1$ **then**
  Initialise to average appearance: $\mathbf{b}^i \leftarrow \mathbf{0}$, $\alpha^i = 1$ and $\beta^i = 0$
else
  Initialise expansion coefficients using mapping function: $\mathbf{b}^i = \mathbf{M}^i \mathbf{b}^{i-1}$
end if
Initialise total parameter vector: $\mathbf{p} = \left[\alpha^i; \beta^i; \mathbf{b}^i\right]$
while !converged($\mathscr{E}_{RLS}$) **do**
  Compute residual vector: $\mathbf{r} = \bar{\mathbf{f}}^i - \hat{\mathbf{\Phi}}^i \mathbf{p}$
  Compute weights: $\{\omega_i\}_{i=1}^{N_{f_i}}$
  Compute spatially coherent weights: $\{\varphi_i\}_{i=1}^{N_c}$ {Equation (4.60)}
  Compute the Hessian matrix {Equation (4.58)}
  Calculate parameter increment $\Delta\mathbf{p}$ {Equation (4.55)}
  Update parameters: $\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}$
end while
Compute lighting invariant features $\mathbf{b}^i$ {Equation (4.56)}
return $\mathbf{b}^i$, α^i and β^i

is captured within the first few modes of variation, utilising the whole gamut of directions of variability will lead to unnecessary computational complexity in evaluating the kernel functions as well as the robust fitting of the appearance expansion coefficients. As such, if rapid fitting is desired, a smaller cutoff fraction of total variation to retain must be employed, in comparison to that of modelling where typically 95-98% of variation is retained. The effect of this severely truncated representation is that the resulting errors for the different elements of the feature vector will exhibit different variance. Furthermore, these errors will generally exhibit some degree of correlation with each other.

Although the use of independent robust scalings in Equation (4.52) ignores the correlations between the feature elements, this approximation is much better than using a fixed scaling factor for all elements. How well this approximation holds and what effect it has on the resulting fitting algorithm will generally be problem dependent. For an arbitrarily truncated representation, the procedure for obtaining the robust weightings is as follows. Initially, when performing PCA on the feature vectors of a particular iteration of the error-bound minimisation procedure, all the modes of variation are be retained[3]. The projection of every feature vector onto the PCA space can then be performed. Given the number of modes to retain, either through manual selection or through a variation retention scheme, the variance for each feature element can be computed by first subtracting the components of the generative model pertaining to the *noise* modes using the previously computed expansion coefficients for each image. Then the variance of the residuals between the generated feature vector using the truncated representation and the true feature vector can be computed independently for each element.

Finally, the choice of which robust penaliser to use in Equation (4.52) should be made

[3]Note that since the dimensionality of the feature vector N_f is generally much larger than the number of available feature vectors N_d, at most $(N_f - 1)$ modes of variation can be found.

such that the exponential of the cost function represents the likelihood of the cropped image as closely as possible. However, since the nature of the outliers is generally unknown, the optimal choice is difficult to deduce. In [119], the performance of the robust normalisation inverse compositional algorithm is evaluated using a number of different robust penalisers. The authors found that the best performing penaliser was that which assumed the derivative takes the form of a Gaussian probability density function:

$$\omega_i = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left\{ -\frac{\mathbf{r}_{(i)}^2}{2\sigma_i^2} \right\}, \tag{4.64}$$

with separate variances for each pixel, where $\mathbf{r}$ is the vector of appearance residuals between the current feature and that predicted by the appearance model.

4.5 Background Invariance

All instances of the iterative error bound minimiser described in Section (4.3) essentially learn the background of the training images, and utilise it to predict LDM parameter updates. This is due to the training set at each iteration that includes feature vectors extracted from perturbed locations, which in many cases includes some background pixels. In applications where the background is predictable, such as in medical image analysis for example, this trait of discriminative fitting is a desirable one, since the boundary between the object and background is a good feature to use in predicting parameter updates. However, in the more general case, learning all possible backgrounds is not viable. As such, fitting algorithms trained on a particular background will perform poorly on images with different backgrounds.

The robust formulation discussed in Section 4.4, can provide a level of invariance towards backgrounds, however, due to the nonlinear estimation of the features to regress, this approach can be expected to be less efficient than its non-robust variants described in Section 4.3. When fitting speed is of the greatest importance, and when no occlusional effects are expected in the image, utilising the robust formulation to account for background variability is less desirable.

In the original AAM formulation [30], background robustness is induced by excluding background pixels from the Jacobian estimation process. For true generative methods, such as those in [83; 104], where the update model is generated directly from background free components (i.e. the mean appearance and their modes of variation), there is no specialisation to any particular background. However, when initialisation is far from the optimum, with a large proportion of the image under the current warp estimate consisting of background pixels, these approaches are prone to terminating in local minima. Recently in [118], background sensitivity was tackled by combining the accurate fitting of AAM's with contour extraction properties of an active contour. Here, the active contour essentially provides the AAM with an initialisation, which includes a minimal amount of background pixels in its feature vector. Although this approach can be utilised to initialise the iterative-discriminative methods described in Section 4.3, it relies on the visual object of interest exhibiting a strong boundary with its background. Furthermore, the approach is somewhat inelegant, requiring a separate procedure to account for the drawbacks of LDM fitting, rather than addressing the drawbacks of the fitting procedure itself.

4.5.1 Invariance through Exclusion

Rather than augmenting the fitting procedure with some peripheral detector, or relying on robust procedures to account for background variability, in this section, a method that leverages on the iterative-discriminative framework is proposed. In order to do so, some of the characteristics of the extracted feature vector, pertaining to background pixel influence, must first be considered:

- Elements of the feature vector that are influenced by background pixels generally correspond to locations in the canonical frame that lie in and around the peripheral of the canonical shape.
- As fitting proceeds and the boundary of the object becomes better estimated, the region in the canonical frame influenced by the background reduces in area.

Considering these characteristics in the context of iterative error bound minimisation, it appears that a strategy of excluding the background influenced pixels from the extracted features, which are then used to predict the parameter updates, is a pragmatic one.

However, in an online setting, there is no easy way of distinguishing background from foreground pixels. Instead, we rely on one more observation, which is that the basin of convergence of most LDM fitting problems is well within half the object's size in parameter deformation. As such, if all elements of the feature vector, which are influenced by background pixels in any of the training instances, are excluded from the regression process, for many problems, the resulting feature vector may still contain sufficiently rich information to make good predictions, in order to reduce the error bound. As fitting proceeds, the number of feature vector elements excluded from update predictions decreases, resulting in a richer information set by which predictions to refine the solution can be made.

If this approach for background invariance is utilised, the size of the feature vector varies between iterations. Furthermore, for implementation with the kernel based non-linear method described in Section 4.3.2, PCA must now be performed solely over the components of the feature vectors that are not affected by the background. The same can be said for the robust method described in Section 4.4. At each iteration, the feature vectors for each training sample are first collected, along with labels for each element of the feature vector, which denote whether the element corresponds to the background. A foreground mask is then built, which retains only features that are labelled as foreground in a large proportion of the sample set, for example 99%. Only features that are covered by the mask are then used to train the update model for that iteration. Using the trained background invariant model for fitting, then, involves the extra step of selecting the feature vector elements that are covered by the mask at each iteration. This procedure can be performed extremely efficiently, since it involves only a binary operation between the feature vector and the mask.

4.6 Conclusion

In this chapter, a new framework for LDM fitting has been proposed. Through the utilisation of an iterative-discriminative paradigm along with the objective of error bound minimisation, the

approach is designed to accommodate fitting for models that exhibit large amounts of intrinsic variability, a problem that is difficult to solve using generative methods. Example prototypes utilising linear and nonlinear regressors are presented along with details of their training and fitting procedures. An extension to handle occluded images is also presented, for an instance of the nonlinear case. Invariance towards unknown and varying backgrounds is also presented, which leverages on the iterative-discriminative framework through a feature selection procedure.

The basic framework, along with its various extensions is designed to address the five goals of LDM fitting. Efficiency is afforded through the utility of simple fixed update models and some approximations for the robust case. Accuracy and reliability are leveraged on the predictive capacity of discriminative methods. Applicability is also afforded through the discriminative framework, which makes no assumptions regarding the LDM's parameterisation, its warping function or the feature vector used. The automaticity of the method relies on the availability of an external crude detector. However various approaches for efficient and accurate object detectors are now numerous, complementing the drawback of locality of the fitting procedures described in this chapter.

Future work on the iterative-discriminative approach will entail improvements to the training procedure of the iterative-discriminative approach. In particular, aspects pertaining to optimal parameter selection, such as the number of iterations, number of weak learners of the asymptotically trained nonlinear method, the regularisation parameter, and the inclusion rate of the background invariant method, all of which must be set manually in the forms presented in this chapter. Efforts to reduce training complexities may also prove to be a worth while endeavour.

Iterative-Discriminative Fitting - Experimental Evaluation

It's not your fault that you're always wrong.
The weak ones are there to justify the strong.

Marilyn Manson

In Chapter 4, a novel approach to LDM fitting, the iterative-discriminative method (ID), was proposed. It leverages on the concept of error bound minimisation, where the error bound over the training data is reduced at each iteration, rather than the more conventional least squares error. This approach was designed to allow simple regressors, with limited predictive capacity, to be utilised. Through a shift in the fitting objective, from directly solving for the optimal step to only reducing the error over worst cases, with the expectation of further reductions down the line, the use of highly sophisticated regressors can be avoided. Furthermore, the error bound minimisation paradigm provides a continuity of objective between the iterations, which favours overall performance over specific instances of the problem.

In this chapter, the efficacy of ID is evaluated in the context of generic face model fitting. This is a difficult problem, which has yet to be addressed adequately in the literature. Most methods tackling this problem have been shown to sacrifice one or more of the five goals of deformable model fitting, outlined in Section 2.3, in order to address some of the others. Through the extensive experiments presented in this chapter, ID is shown to be a powerful general technique, which makes significant inroads into solving the problem of generic face fitting. Not only does it exhibit excellent generalisability, its overall fitting accuracy is superior to a number of existing methods for LDM fitting. Furthermore, the significant performance improvement is attained without sacrificing computational efficiency.

In Section 5.1, the general experimental framework is discussed, where a number of baseline methods, used in a comparative setting with the various prototypes of ID, are outlined. The applicability of linear regressors in ID is evaluated in Section 5.2, where both variants of the training procedure, described in Sections 4.3.1 and 4.3.1, are assessed for merit. The extension of the approach to nonlinear regressors, utilising the novel Haar-like feature based regressor described in Section 4.3.2, is then evaluated in Section 5.3. The ability of ID to handle occlusions, addressed through a robustification of the kernel-based nonlinear approach, outlined in Section 4.4, is evaluated in Section 5.4. Finally, the background invariant extension, described in Section 4.5, is assessed in Section 5.5. This chapter concludes in Section 5.6 with a summary of the experiments conducted and a discussion on directions of future work.

5.1 Experimental Setup and Baseline Methods

The methods described in Chapter 4 are unique in that they do not depend on the type of warping function, the feature vector used or even the parameterisation of the deformations. This makes them applicable for many instantiations of model based fitting. Since LDMs form the focus of this thesis, in order to evaluate ID's performance, one of the more popular flavours of LDMs, the Active Appearance Model (AAM), is used as a prototype. First proposed in [43], the AAM utilises a piecewise affine warp to crop the image at the current parameter settings (see Section 3.3.3). It models the object of interest as a 2D visual object, composing the linear intrinsic shape and appearance variations with global transformation functions (a similarity transform for shape and a linear lighting model for appearance), to project the model onto the image frame.

There are numerous approaches to AAM fitting, most of which stem from a generative perspective. To evaluate the ID's performance, it is compared against five existing methods for AAM fitting, namely:

The Fixed Jacobian Method (FJ), first proposed in [30], uses the combined appearance representation, which accounts for correlations between the intrinsic shape and appearance parameters (see Section 2.1.3). It uses the normalised appearance residual feature to drive the fitting, where it is assumed that the Jacobian of the appearance residuals is fixed for all settings of the model parameters. Since this assumption holds only loosely, the method requires the use of an adjustable step size, where at each iteration the predicted parameter updates are continually halved until a reduction in the appearance difference between the model and the cropped image is attained. The method affords reasonable fitting speeds by virtue of its fixed Jacobian assumption, with the main bottleneck being the appearance generation procedure.

The Project-out Inverse Compositional Method (POIC), first proposed in [83], adapts the inverse compositional framework in [58] for use in AAMs. The generative cost function, which assesses fitness through the difference between the model's appearance and the cropped image, is grouped into two components: one that lies within the subspace of appearance deformations, and another one that is orthogonal to it. As such, the procedure requires optimisation over the shape parameters alone, assuming the optimal choice (in a maximum likelihood sense) of the appearance parameters is chosen at each iteration. Since the derivatives are computed in the canonical frame, by virtue of its inverse compositional framework, most of the problem's computation needs to be performed only once at training. As such, the fitting procedure affords an extremely rapid evaluation, requiring only a matrix-vector multiplication, without the requirement to compute the model's appearance explicitly. The approach affords similar computational efficiency to the shape based AAM proposed in [29]. However, the use of a fixed linear update model is better justified in POIC, since it is derived analytically from a generative perspective.

The Simultaneous Inverse Compositional Method (SIC), which was proposed in [4] for general image alignment, and evaluated as an AAM fitting procedure in [54; 91], directly solves the appearance residual cost function in the canonical frame. Although the derivative of the warping function can be precomputed, unlike POIC, the linear update model

cannot, since it relies on the current appearance parameters. As such, SIC can be very computationally demanding, especially when the visual object exhibits large amounts of variability, reflected through the number of shape and appearance modes. An example of this is in the generic face fitting problem with which this chapter is concerned. As such, for the purpose of comparison with ID, the implementation used in this chapter is the efficient approximation of SIC, where the linear update model is built using the assumption that the appearance parameters are fixed. However, rather than evaluating the update model using the current estimate of the appearance parameters, as proposed in [4], it is evaluated at the mean appearance (i.e. all intrinsic appearance parameters are set to zero), allowing the update model to be precomputed. The idea of using the current estimate of the appearance parameters to build the update model is only applicable when the current estimates are close to their true values. Since this chapter deals with model fitting rather than tracking, in which case the appearance parameters from the previous frame may be close to their values in the current frame, building the linear update model using the current parameter estimates will generally be a poor approximation. Furthermore, extracting the initial appearance estimates from the initial cropped image will also be inaccurate, since the appearance model will essentially fit to appearance variations caused by misalignment rather than intrinsic variations in the visual object's appearance. As such, when assuming the variations in the visual objects appearance is Gaussian, the optimal choice for computing the update model is the mean image, which on average is closest to all instances of the visual object. With this approximation, SIC also affords rapid fitting. However, compared to FJ, it is less efficient since it uses an independent representation of variability (i.e. modelling shape and appearance separately).

The Normalisation Inverse Compositional Method (NIC), also proposed in [4], uses the inverse compositional parameter update model for template matching (the template here being the mean appearance), which is applied to the mean subtracted cropped image, normalised with respect to the directions of appearance variability. Compared to POIC, this approach requires the extra steps of: (1) Projection of the error image (mean subtracted cropped image) onto the subspace of appearance variations, (2) Generating the model's appearance from the projected coordinates in the space of appearance variations, and (3) Subtracting the generated appearance from the error image. As such, not only is the approach slower than POIC, it is also slower than the efficient approximation of SIC described above. However, as will be seen through the experiments in this section, NIC has the ability to outperform the other variants of the inverse compositional approach on a generic face database.

The Robust Inverse Compositional Method (RIC), first proposed in [53], robustifies the inverse compositional method through the use of a robust penaliser. The method, which is based on NIC, replaces the least squares fitting criterion with an M-estimator (robust penaliser), leading to an iteratively reweighted least squares fitting procedure. A reduced computational complexity is attained by assuming a degree of spatial coherence of the outliers, where errors in each triangle of the piecewise affine warp are weighted equally.

All five of these baseline methods are implemented in the C++ deformable model library `DeMoLib`, which was developed as part of this dissertation (see Appendix C). Apart from the

Table 5.1: Appearance Model Details for 4-fold Cross-validation

Experiment	P	M_s(95%)	M_t(95%)	M_c(98%)
1	34092	19	100	78
2	28267	20	99	77
3	28244	20	101	79
4	28086	20	97	75

P, M_s, M_t and M_c as described in Section 2.1

five methods described above, there exists a large number of other methods for AAM fitting. However, most of these focus on feature vectors and image processing techniques, to make the assumption of linearity better justified, rather than on formulations of the fitting procedure itself. An overview of other existing methods is presented in Section 2.3.

All experiments in this chapter are performed on the IMM Face database [89], which is described in Section 3.6.1. To evaluate the non-robust methods, the database was divided into 4 equally sized parts, where each subject is contained in only one part, separating subjects used for training and testing. A 4-fold cross validation was then performed on each method, training on data from three parts and testing on the remaining part. This procedure was repeated four times, utilising different training and test sets in each. For the baseline methods described above, the models were trained on three levels of a Gaussian pyramid to help avoid local minima in their generative cost functions. The same shape and appearance models were used in all methods, the details of which are given in Table 5.1 for each of the four subdivisions at the lowest pyramid level. In each case, 95% of the total variation in shape and appearance and 98% of the combined appearance variation was retained.

To evaluate the fitting performance of the baseline methods, the AAM parameters were randomly perturbed from their optimal settings, 100 times in each test image, within $\pm 10^o$, ± 0.1, ± 20 pixels, and ± 1.5 standard deviations of rotation, scale, translation and non-rigid shape parameters, respectively. These ranges were chosen to mimic the initialisation capacity of a generic face detector. Each method was then iterated to convergence, or a maximum of 20 iterations per pyramid level. The combined results of the 4-fold cross validation experiments are presented in Figure 5.1, where the convergence rates, average accuracy of converged trials and fitting times are shown in the legend. Here, convergence is declared if the final point-to-point RMS error is smaller than at initialisation (i.e. the algorithm does not diverge). A similar measure of convergence has been used in the extensive experiments, presented in [8; 6; 4; 5]. The reported fitting times were obtained from running the code, implemented in the C++ library `DeMoLib` (see Appendix C), on a 3GHz machine with 1GB of RAM, and do not include the time taken to build the Gaussian pyramids.

From these results it is clear that FJ is the most stable, affording a 79.73% convergence rate. It also affords the best average convergence accuracy at 6.41 point-to-point RMS error. Although it exhibits a slower fitting time than POIC, it is significantly more efficient than either SIC or NIC. Out of the three variants of the inverse compositional method, POIC achieves the best average convergence accuracy. However, its convergence rate is much poorer than the other methods, affording only 51.93% converged samples. This is in line with the results presented in [54], where it is argued that POIC is suitable only for person specific models.

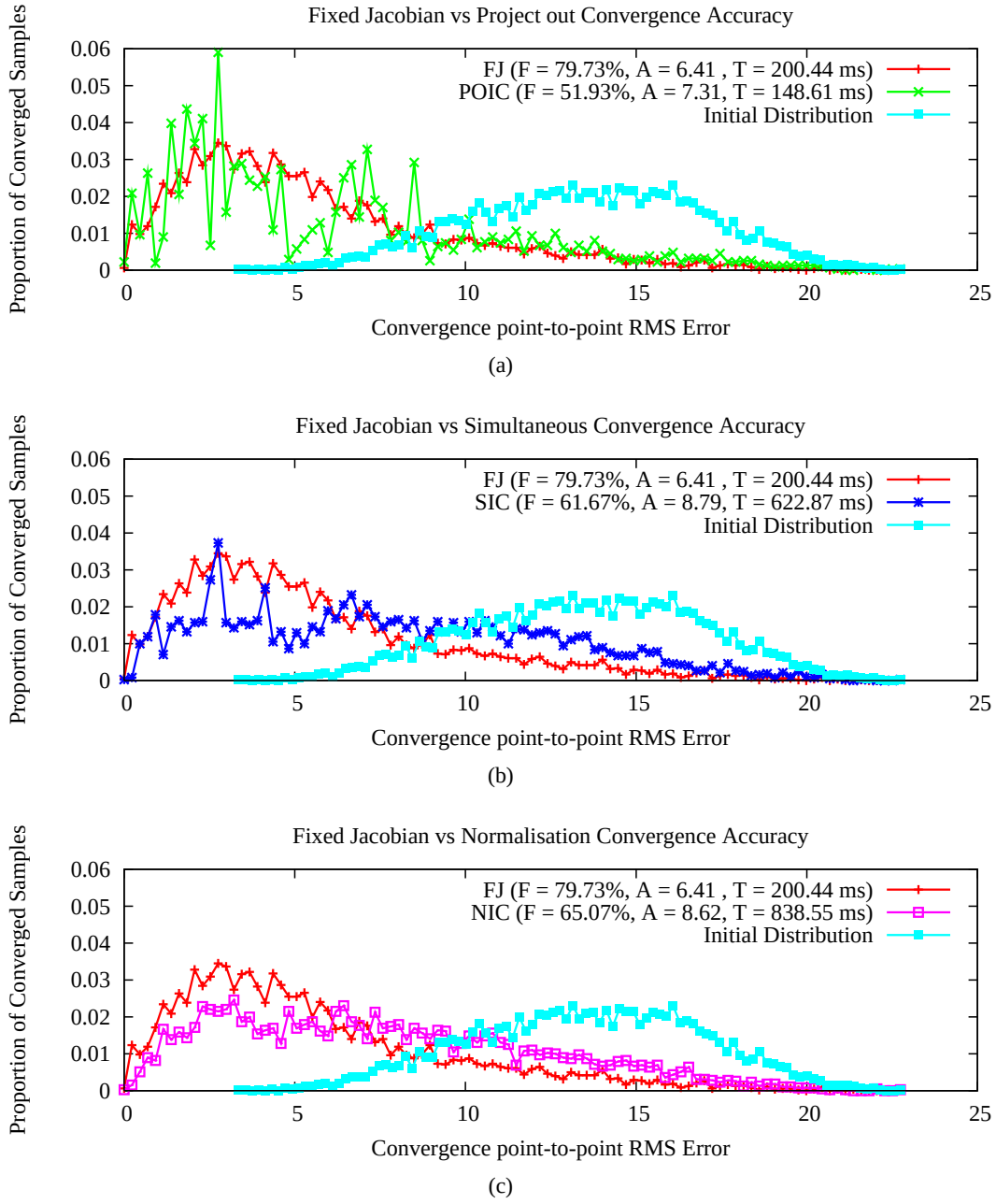


Figure 5.1: Performance comparisons between the non-robust baseline AAM fitting methods. **(a):** FJ vs. POIC. **(b):** FJ vs. SIC. **(c):** FJ vs. NIC. In the legend, "F" denotes the convergence rate, "A" denotes the average accuracy of converged trials in point-to-point RMS error, and "T" denotes the average fitting time of converged trials. The histograms were built, only from samples that converged.

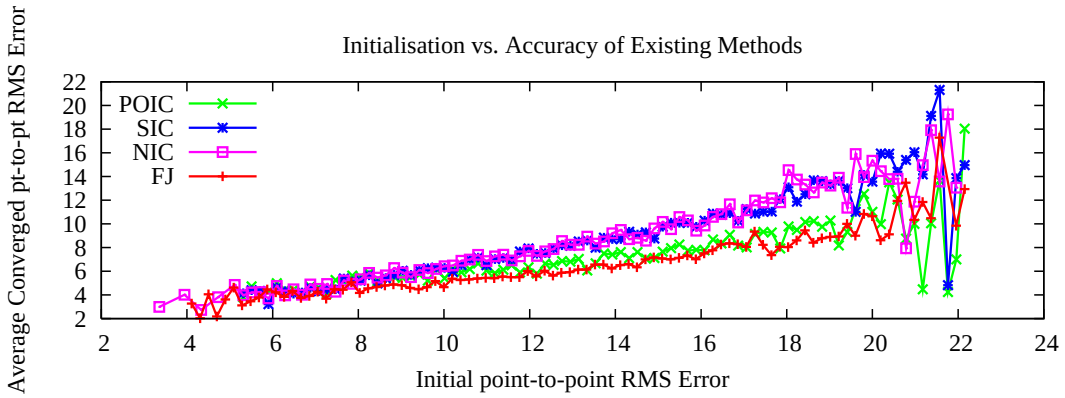


Figure 5.2: Effects of initialisation on convergence accuracy on FJ, POIC, SIC and NIC.

Although SIC affords a convergence rate improvement over POIC, it still performs poorly, affording only 61.67% convergence. It is suspected that this is related to the large number of parameters involved in the optimisation, resulting in a higher likelihood of getting trapped in local minima than FJ (note that SIC uses an independent appearance model). Although the true SIC implementation may improve results here, the fitting times required by this method are impractical for the size of the models used in generic face fitting (see Table 5.1). Even the efficient approximation used in these experiments is around three times slower than FJ, requiring on average 622.87ms to fit an image. From a small number of separate experiments, it was found that the full SIC implementation can further increase fitting time by a factor of ten or more. Finally, a small improvement in convergence rate and accuracy over SIC is afforded by NIC. However, this is achieved at the cost of higher computation times.

In Figure 5.2, the effects of initialisation on the accuracy of converged samples is shown. The plot shows a clear trend of performance deterioration as the model is initialised further from its optimal settings. This trend is exhibited by all four non-robust baseline methods and is a characteristic typical of generative fitting regimes, which tend to terminate in local minima, despite their application, here on a Gaussian pyramid. As will be seen in the sections to come, this is one area where ID comes into its own, affording good performance over the whole range of initialisations.

In conclusion, out of the four non-robust baseline methods evaluated in this section, FJ performs the best, both over convergence accuracy and frequency. Although it exhibits a slower fitting time compared to POIC, it is still significantly faster than either SIC of NIC. As such, in the following sections, the various flavours of the ID approach are compared with FJ exclusively.

5.2 Linear Fitting

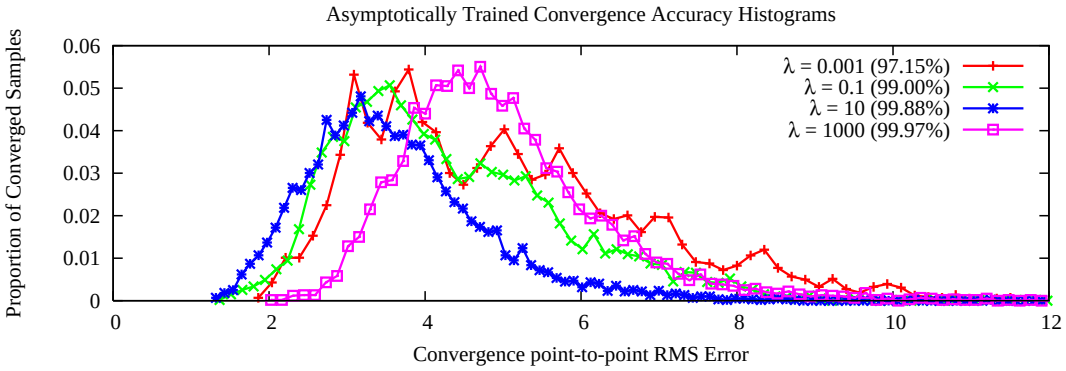
The first variant of ID is that which utilises a linear regressor to update the AAM parameters (see Section 4.3.1). There are two options for the cost function to be used to train the method. For the asymptotically penalised cost function (see Section 4.3.1), which will be subsequently referred to as the asymptotically trained linear iterative-discriminative method (ATLID), the

Figure 5.3: Examples of the normalised raw cropped image feature used in the linear iterative-discriminative approach.

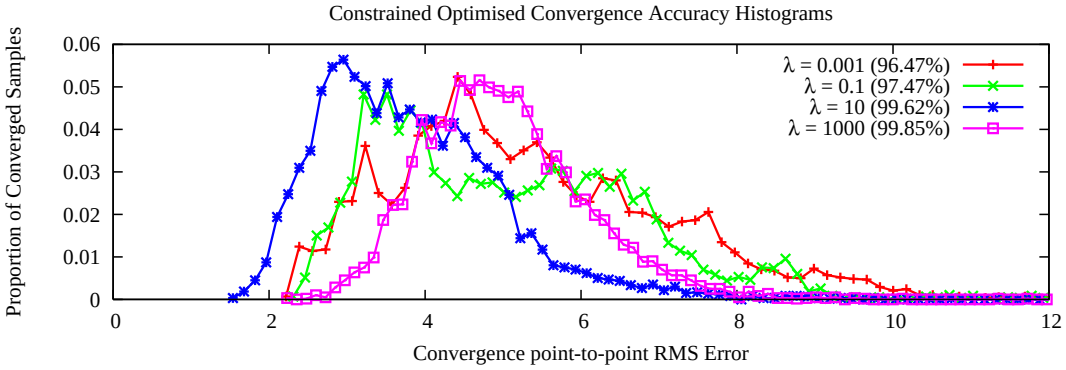
limited memory BFGS algorithm (L-BFGS) [74] was used to optimise the regressor for each AAM parameter, independently of all others, in each iteration. For the error bound constrained cost function (see Section 4.3.1), which will be subsequently referred to as the constrained optimisation trained linear iterative-discriminative method (COTLID), the `libsvm` library [24] was used to solve the ν-SVR problem for each AAM parameter.

As with the baseline methods described above, a 4-fold cross validation procedure was used to evaluate the two linear methods. In each case, the training set consisted of feature vectors obtained by perturbing the AAM parameters from their optimal settings within the ranges described in Section 5.1. For all experiments in this section, a sample size of $N_d = 2000$ was used at each iteration. As for the feature vectors themselves, the normalised raw cropped image was used (see Equation (4.4)), examples of which are shown in Figure 5.3. The advantage of using these features is that they can be obtained extremely rapidly, requiring only a warping process followed by a normalisation procedure, centering the average feature values at zero, and scaling to a standard deviation of one. Finally, it should be noted here that in order to reduce training times, the canonical shape used in these methods, which defines the size of the feature vectors, is scaled down by one half of that used in the baseline methods in Section 5.1. However, the model's fitting is still performed on the original image. Although better performance may be obtained by using the full scaled model, since more information would be available to make the parameter update predictions, the training time was deemed impractical for the experiments in this section. Even the scaled down version required around eight hours of training for each of the models.

In both variants of the linear ID method, a suitable choice of the regularisation parameter λ must be set manually. In Figure 5.4, the combined results of the 4-fold cross validation experiments on both methods using four different settings of λ are presented, where the model was trained and fitted with 10 fitting iterations. Here, the rate of convergence of each trial is shown in the legend. Notice that the generalisability of both methods improves as λ is increased, as can be seen through the convergence rates. This is to be expected as larger values of λ promote *simpler* solutions through the selection of a regressor with smaller L_2-norm. Examining the histograms of the converged samples, an initial improvement from $\lambda = 0.001$ to $\lambda = 10$, followed by a deterioration with a further increase to $\lambda = 1000$ is noticable. When λ is chosen at too small a value, the training procedure under-regularises the regressors, leading to reduced generalisability. However, when chosen too large, the regressors become over-regularised, restricting their predictive capacity. As such, the results here highlight the



(a)



(b)

Figure 5.4: Convergence performance of the linear method trained on four settings of the regularisation parameter λ on the IMM Face database. **(a):** results of a model trained using the asymptotically penalised objective. **(b):** results of a model trained using the linear ν-SVR method. Convergence rates of each trial are shown in the legend.

importance of selecting the most appropriate λ in order to maximise the capacity of the linear update model. Further improvement may be obtained by performing similar experiments on closer spaced values of λ, however, due to the lengthy training time involved, this was not pursued here.

In Figure 5.5, the performances of the two variants of the linear ID method, trained with $\lambda = 10$, are compared against each other, along with the FJ method. An examination of Plot (a) shows that, although ATLID exhibits a slightly better convergence rate, at 99.88% compared to the 99.62% of COTLID, the accuracy of its converged trials is inferior to that of COTLID. This is expected, however, since the ATLID requires that the error bound over *all* samples is reduced, rather than merely a large fraction of samples, as in COTLID. In these experiments, COTLID was trained with $\nu = 0.001$, which places the error bound at 99.9% of the samples. Compared to FJ, both ATLID and COTLID perform significantly better, both in convergence rate as well as in convergence accuracy. In fact, the average converged accuracy of the two methods is about twice as good as that of FJ. The effects of initialisation on the convergence accuracy of ATLID, COTLID and FJ can be seen in Plot (b). From this, the reason for the sig-

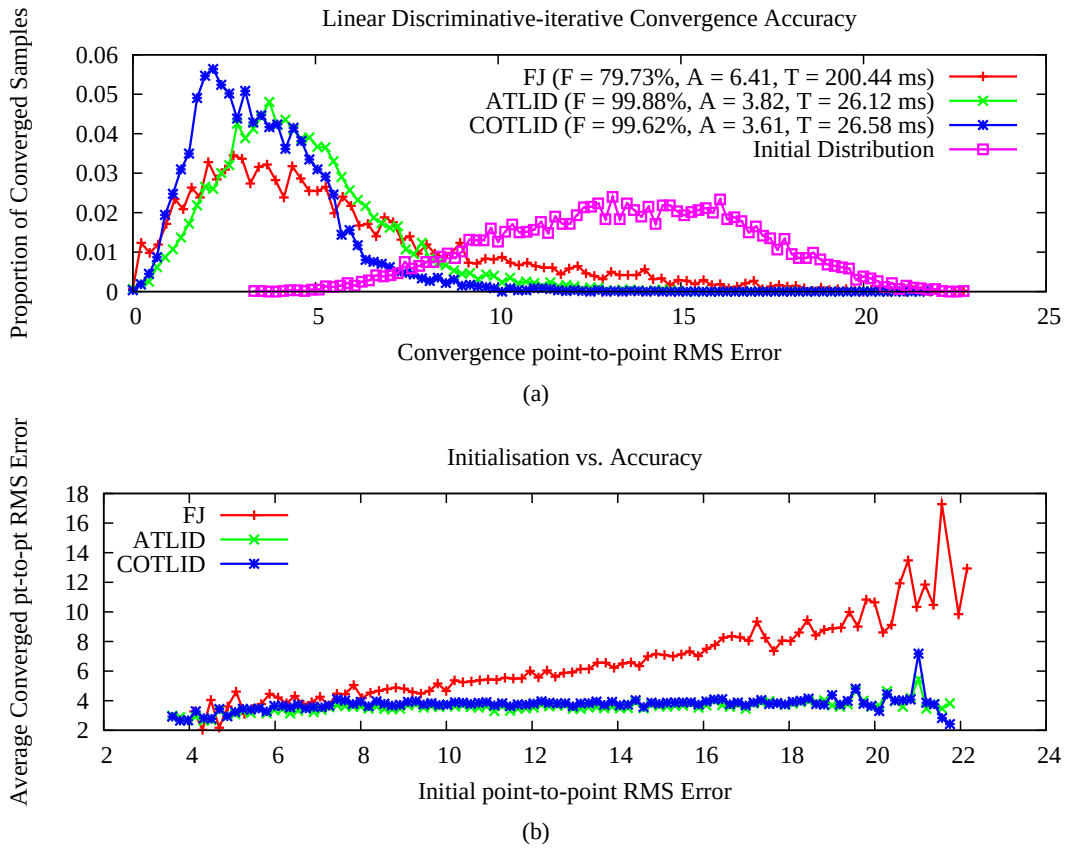


Figure 5.5: Performance comparison between ATLID, COTLID and FJ. **(a):** convergence accuracy histograms. **(b):** effects of initialisation on convergence. In the legend, "F" denotes the convergence rate, "A" denotes the average accuracy of converged trials in point-to-point RMS error, and "T" denotes the average fitting time of converged samples. The histograms were built only from samples that converged.

nificant performance improvement achieved by ATLID and COTLID becomes apparent. The average convergence accuracy of FJ deteriorates the further initialisation is from the optimum, due to its generative framework, which tends to encounter local minima before it reaches the global one. In contrast, the deterioration of ATLID and COTLID, which is based on a discriminative framework, is not as dramatic, maintaining a good average convergence accuracy up to a capture range of around 20 pixels point-to-point RMS, which corresponds to the limit of the pertubations used in training. It should be noted, however, that FJ does exhibit a higher proportion of converged samples with very small fitting errors (see Plot (a)). This can be attributed to FJ's generative fitting regime, where given good initial conditions, and the approximation regarding the fixed Jacobian is reasonable (i.e. the subject has similar shape and appearance to the mean of the model), then highly accurate fitting can be expected. ATLID and COTLID, on the other hand, are specifically trained to attain good *overall* performance.

Despite the significant improvement in the performance of ATLID and COTLID compared to FJ, it is afforded without sacrificing computational efficiency. Both methods, which afford similar computational costs, fit an image in around one-tenth the time it takes FJ. The

significant computational savings can be attributed to thee factors: (1) The normalised raw appearance feature is much cheaper to evaluate than its appearance residual counterpart, (2) No step size adaptation is required in fitting, and (3) The model is trained and fitted with 10 iterations only. In fact, even compared to POIC, perhaps the most efficient fitting procedures to date, the fitting times of ATLID and COTLID are still significantly better (see Figure 5.1).

In conclusion, the linear update model serves as an excellent regressor for ID. Full advantage is taken of its limited capacity by the iterative error bound minimisation framework, allowing high convergence rates and overall accuracy, whilst affording what is perhaps one of the most efficient AAM fitting regimes to date. The good performance reported here was achieved on the highly challenging task of *generic person fitting*, where previous methods have sacrificed fitting efficiency in order to tackle such a task. This method does exhibit the drawback of extended training times. However, when training time is of no concern, this method provides an excellent substitute for current AAM fitting regimes.

5.3 Haar-like Feature Based Fitting

In Section 4.3.2, two variants of ID were proposed, which utilise nonlinear regression functions. The first method utilises a novel regressor based on the Haar-like features [72]. In the second method, a kernel-based regressor is used, which takes, as its features, the coordinates in a reduced subspace of the cropped image. In this section, we will be concerned with the first method exclusively, which will be referred to as the Haar-like feature based iterative-discriminative Method (HFBID). The second method involves a large number of design parameters, which include the regularisation parameter as well as those that define the kernel function. These parameters must be selected heuristically or through a cross validation procedure. As such, due to the extended training times involved, evaluation of the kernel-based nonlinear ID method will not be pursued here, deferring it to a future study. However, the framework of the kernel-based method serves as a basis for the formulation of the robust ID method, which was developed in Section 4.4, and is evaluated in Section 5.4

To evaluate the performance of HFBID, again, the 4-fold cross validation procedure, outlined in Section 5.1 was utilised. The training data was obtained in a similar manner to that described for the experiments in Section 5.2, utilising a sample size of $N_d = 2000$ at each iteration. The Haar-like based features described in Section 4.3.2 were evaluated on the raw cropped image feature. In order to build the summed area tables (SAT), the raw cropped image is placed inside a rectangle that fits the canonical shape exactly. Pixels within the rectangle that are not within the convex hull of the canonical shape are set to zero. As with the linear fitting experiments in the previous chapter, here the scaled down version of the canonical shape is used to reduce training time. To encourage invariance to global lighting effects, the normalised Haar-like features are employed, requiring two SATs to be built (see [72] for details). In fact, since the extended Haar-like features are used here, two pairs of SAT images are built: one for upright features and another for rotated features.

Figure 5.6 shows the distribution of the training samples at different stages of the training process. Plots (a), (c) and (e) illustrates the capacity of the weak function set, described in Section 4.3.2, to significantly reduce the spread of these samples despite the relative small value of N_d, which amounts to a very sparse sampling of $\mathcal{H}$. From Plot (b), (d) and (f) it is

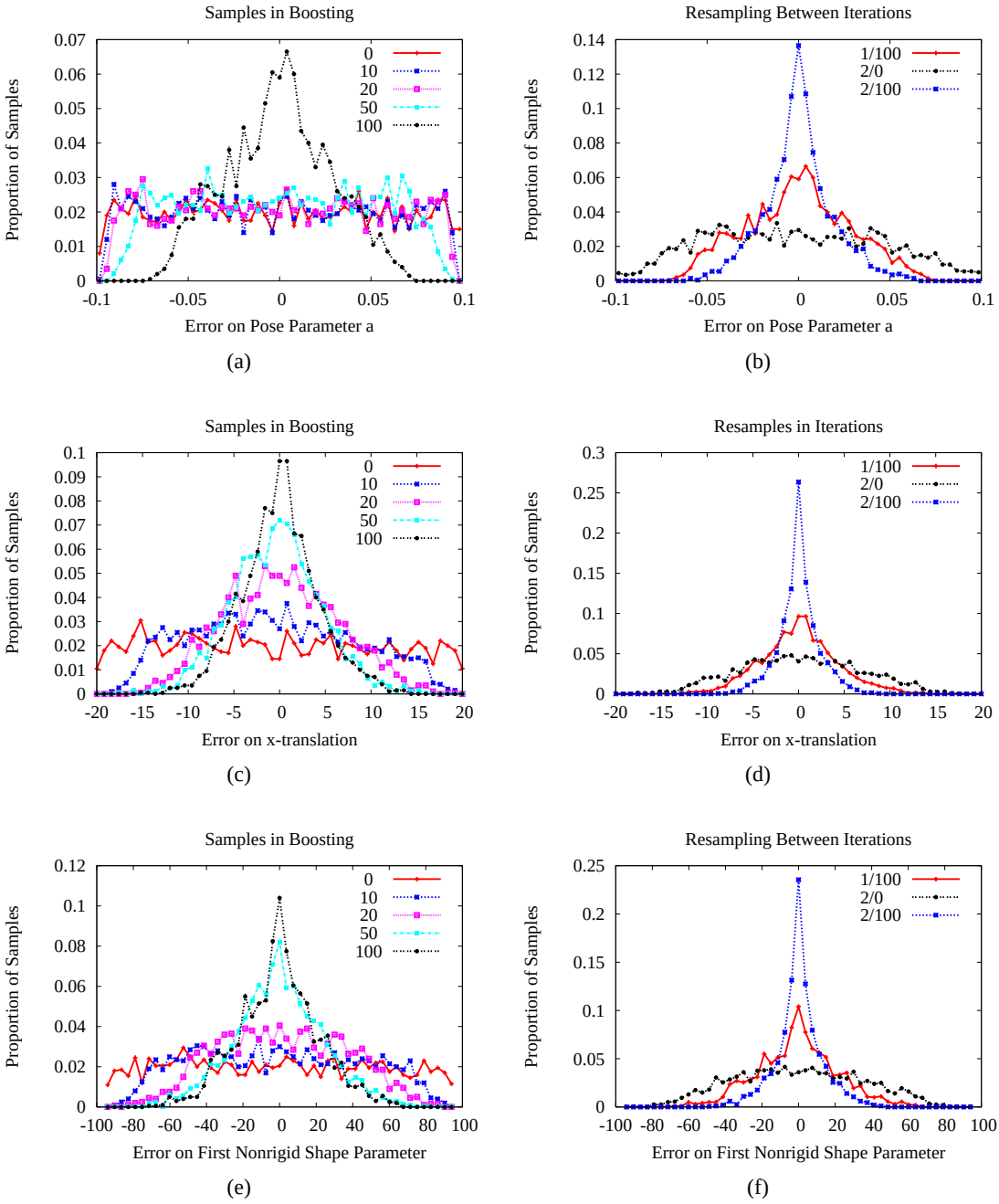


Figure 5.6: Distribution of the training samples of the IMM database throughout HFBID's training process on the pose parameter a, the x-translation parameter and the first nonrigid shape parameter. **(a), (c) & (e):** Redistribution of samples about the optimum as weak learners are added to the ensemble of the first iteration. Legend denotes the number of weak learners in the ensemble. **(b), (e) & (f):** The effect of resampling between iterations. Legend denotes (iteration)/(number of weak learners in ensemble of that iteration).

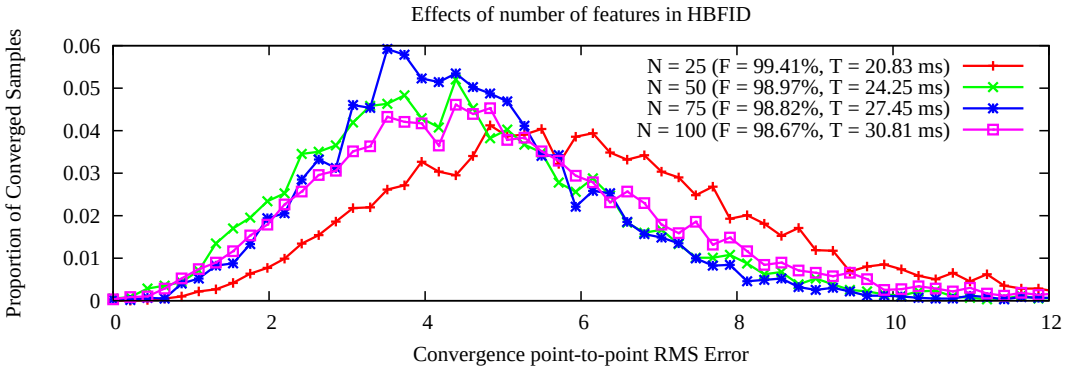


Figure 5.7: Convergence performance of HFBID, trained at four different numbers of features N. The symbols "F" and "T" in the legend denote the convergence rate and the average fitting times, respectively.

clear that with the modest training set size used, the boosting process by itself significantly overlearns the data, as shown by the *spreading-out* of the resampled data in the next iteration. However, this artifact of the boosting process is more than compensated for in the next iteration, where the final distribution is even less spread than its predecessor. This trend is continued throughout the iterations and for all parameters.

HFBID requires three free parameters to be set: the shrinkage factor η, the number of features for the regressors in each iteration N, and the number of features to evaluate before choosing one to be added to the ensemble N_t. In the experiments presented here, a shrinkage factor of $\eta = 0.5$ is used in all cases. To increase the likelihood of selecting the optimal feature to be added to the ensemble at any stage of the boosting procedure, the value of N_t should be as large as possible. However, increasing N_t also increases the training time of HFBID. For all experiments in this section, $N_t = 200$ was chosen as a good compromise between training time and model quality.

The effects of varying N were investigated by performing cross validation experiments on the method, trained at four different settings of N. Due to time constraints, only one sub-group of the IMM Face database was evaluated here. The results of these experiments are presented in Figure 5.7, where the convergence rate and fitting times are shown in the legend. From these results, it is clear that increasing N deteriorates the generalisability of the method, as can be seen through the reduction of the convergence rate. Furthermore, increasing N also increases the computational complexity of the method, as the regressors contain more weak learners to evaluate. On average, an increase of around 0.1ms in fitting time results from adding an additional feature to the regressor. In terms of accuracy, utilising $N = 50$ provides a significant improvement over $N = 25$. However, increasing N further only deteriorates the accuracy as the method, as it overlearns the training data.

To compare HFBID against the other methods discussed so far, a 4-fold cross validation experiment was conducted, setting $N = 50$ in each case. The results of all four experiments combined are presented in Figure 5.8, where the COTLID and FJ methods are also shown for comparison. The convergence rate, average accuracy of converged samples, and fitting times are shown in in the legend. From these results it is apparent that HFBID exhibits significantly

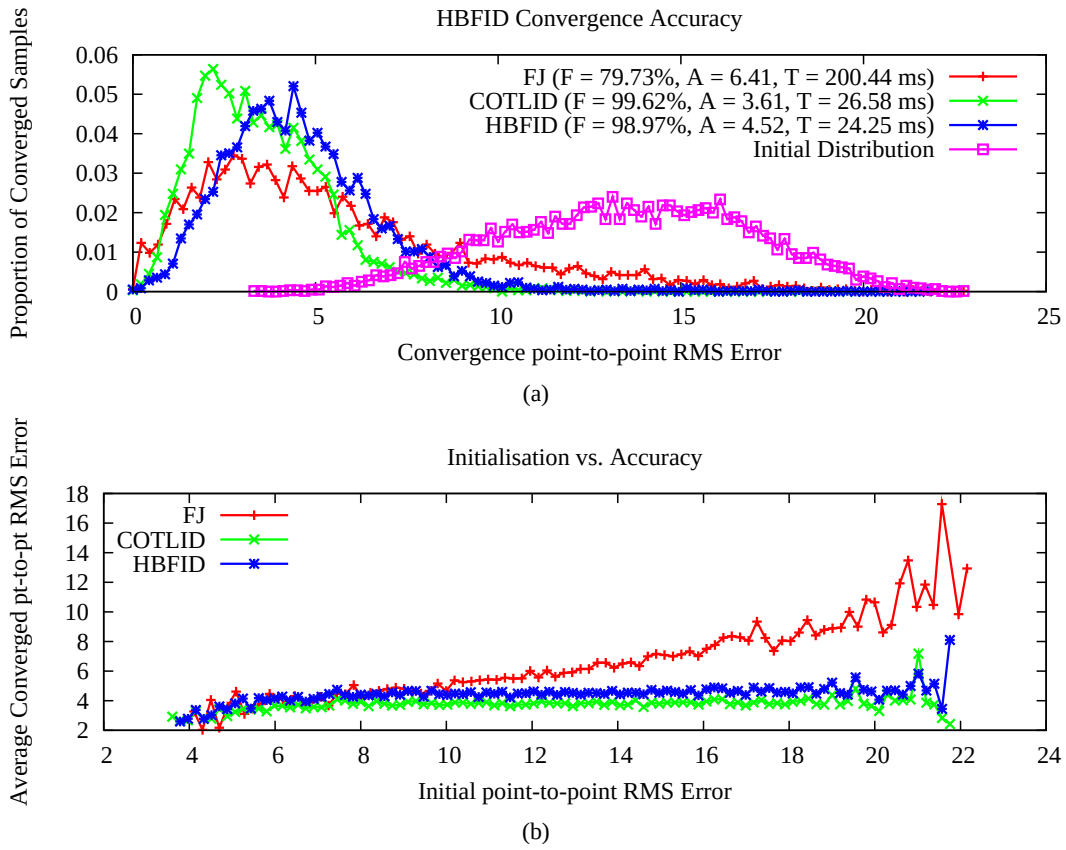


Figure 5.8: Performance comparisons between HFBID, COTLID and FJ. **(a):** convergence accuracy histograms. **(b):** effects of initialisation on convergence. In the legend, "F" denotes the convergence rate, "A" denotes the average accuracy of converged trials in point-to-point RMS error, and "T" denotes the average fitting time. The histograms were built only from samples that converged.

better performance than FJ. However, it is outperformed by COTLID in all respects. Although the difference in average fitting time and convergence rate of the two methods was marginal, HFBID failed to achieve the same fitting quality as COTLID. A possible cause for this may be the small sample size N_t, from which a feature is selected to be appended to the ensemble. This results in a very sparse sampling of the space of possible features, which leads to trained ensembles that are sub-optimal. Finally, examining Plot (b) in Figure 5.8, HFBID exhibits similar behaviour to COTLID, maintaining a good average convergence accuracy up to around 20 pixels initial point-to-point RMS error, albeit with slightly larger errors.

In conclusion, although nonlinear regressors can potentially provide more accurate predictions than their linear counterpart, their training procedure is generally more complicated. In the case of HFBID, only a local solution for the predictor can be attained due to the greedy learning properties of the boosting procedure used to train the method. Furthermore, a true implementation of the boosting procedure, which requires the evaluation of all possible features before appending one to the ensemble, is not computationally feasible for most problems, due to the large number of possible Haar-like features. Due to these training complexities, in

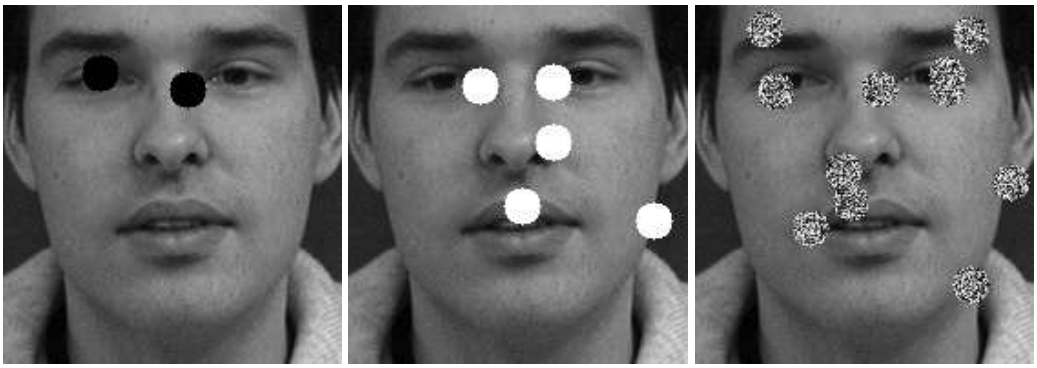


Figure 5.9: Examples of synthetically occluded images, used for evaluating RID. **Left:** black occlusions of 5% of landmarks. **Middle:** white occlusions of 10% of landmarks. **Right:** random occlusions of 20% of landmarks.

practice, the simpler linear models may be a more effective regressor to use within the ID framework.

5.4 Robust Fitting

In Section 4.4, ID is made robust, utilising kernel-based regressors on observations with reduced dimensionality, as presented in Section 4.3.2. The main idea is to perform robustification at the feature extraction stage, making the effects of occlusion or unseen appearance variations transparent for the remainder of the fitting procedure. This is achieved by performing generative appearance fitting on the observations, utilising a reduced linear subspace representation. In order to reduce the computational complexity of the robust feature extraction procedure, two measures are taken. First, the outliers are assumed to exhibit a degree of spatial coherence, similar to that proposed in [55]. Secondly, a linear mapping function is utilised to obtain good initialisation of the appearance parameters, relating their values between consecutive iterations, to encourage rapid convergence of the robust feature extraction procedure.

To evaluate the efficacy of the robust iterative-discriminative method (RID), the linear kernel was used, since it requires only the regularisation parameter to be set manually. Most nonlinear kernels require the manual setting of one or more kernel parameters, for example the kernel width of the radial basis functions, which may be difficult to select without a cross-validation procedure. It should be noted here that the performance of RID using a linear kernel can be expected to be poorer than the non-robust linear methods evaluated in Section 5.2. This is because the linear regressors take, as input, a feature vector of reduced appearance subspace coordinates, rather than the observations themselves. As such, the prediction space of linear RID is more restricted than that of its non-robust counterpart, which performs predictions directly on the whole observation vector. However, since the utilised features represent the main directions of variations of the perturbed appearance space, reasonable performance can still be expected. As discussed in Section 4.4, this is because the major directions of the perturbed appearance subspace mainly correspond to variations caused by misalignment, rather than the visual object's intrinsic appearance variations. For all experiments in this section, the feature

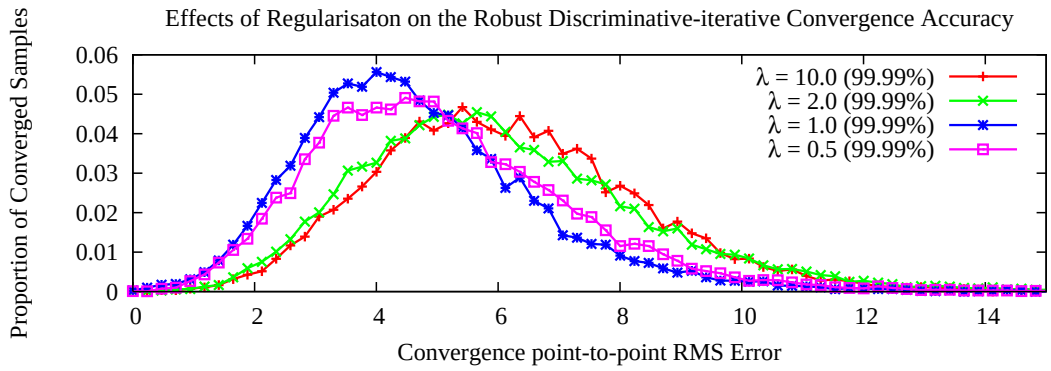


Figure 5.10: Convergence performance of RID, trained on four different settings of the regularisation parameter λ. Here, the images used for training and testing are the same, albeit with different perturbations.

vectors are obtained by applying PCA to the raw cropped image, with truncation placed at 90% of the total variation of the samples.

Since the IMM Face database contains no images with occlusions, two sets of experiments were conducted to investigate RID. In the first, the database was divided into two groups, where the first group consisted of images of subjects with facial hair (such as a moustache and/or beard), and the other of subjects without any facial hair. RID was then trained on the second group, leaving the first group for testing. A similar approach for evaluating robust AAM fitting was presented in [99], where facial hair, which is not modelled in the trained fitting procedure, is used to represent real outliers. In the second set of experiments, occlusions are generated synthetically on images in the second group (i.e. the group on which the fitting procedure is trained). The synthetic occlusions consist of filled circles with a radius of 10 pixels, placed at random landmark locations and filled with either black, white or uniformly sampled random values, in order to evaluate the effects of different occlusion types on fitting performance. Experiments were then conducted on images with 5%, 10% and 20% occluded landmarks. Examples of the synthetically occluded images are shown in Figure 5.9.

Since the variant of RID evaluated here utilises a linear kernel, the training procedure requires only the regularisation parameter λ to be chosen. In Figure 5.10, the effects of varying λ on the performance of RID is shown. In each case, the training procedure involved $N_d = 2000$ samples. Note that the results here are obtained by evaluating the fitting procedure on the *same* images as it was trained on, hence the uncharacteristically high convergence rate[1]. As can be seen, the effects of varying λ in RID is similar to that on ATLID and COTLID in Section 5.2, and HFBID in Section 5.3. Choosing a value for λ that is too large leads to over-regularisation, resulting in poor accuracy, and choosing too small a value leads to poor generalisability, due to under-regularisation. It should be noted here that due to ID's resampling procedure, choosing an unsuitably small λ does not necessarily lead to highly accurate predictions over the training set, since, although the same images are used in training and testing, the samples of AAM observations obtained at perturbed settings are different. As such, under-regularisation exhibits

[1]In the ν-SVR method used to train the liner regressors, a value of $\nu = 0.001$ was used which guarantees that at least 99.99% of the samples lie within the error bound.

reduced generalisability on the training set also. This effect can be seen by examining the effects of decreasing λ from 1.0 to 0.5 in Figure 5.10.

In order to obtain a comparative measure of RID's performance, it was compared with RIC (see Section 5.1). The implementation used here closely follows that described in [55]. An important aspect of RIC, not mentioned in of any publications dealing with it (see [5; 55; 119]), is the effect of using the spatial coherence assumption regarding outliers on the convergence rate of the algorithm. Since the robust weights applied to the Hessian and the gradient are not commensurate, through experimentation it was found that the parameter updates predicted by RIC consistently underestimate the step size. The result of this is a slow convergence rate when the true weights are used to build the gradient, but the spatially coherent weights are used to build the Hessian. An example of this is illustrated in Figure 5.11, where two cases are considered:

- The spatially coherent robust weights are used to build both the Hessian and gradient in each iteration (C-RIC).
- The spatially coherent robust weights are used only to build the Hessian, with the true robust weights used for the gradient as described in [55] (NC-RIC).

As can be seen, C-RIC affords a significantly faster rate of convergence than NC-RIC, finding a minimum after 67 iterations, with a large proportion of the error reduced after only 20 iterations. NC-RIC, on the other hand, did not reach convergence, even after 200 iterations. The slow convergence of the NC-RIC can be attributed to the way in which the spatially coherent weights are chosen as the average over a region. From the effects this has on the convergence rate, it is clear that the sum of the true robust weights in a particular region is smaller than the sum of N_R copies of the average robust weight, where N_R is the number of pixels within that region. As such, by using non-commensurate weights, NC-RIC selects updates closer to the gradient direction rather than the desired Gauss-Newton update, leading to slow convergence. Finally, it should be noted that RID also exhibits a similar characteristic to RIC in this respect, since it uses the same assumption regarding spatially coherent outliers.

In Figure 5.12, the performance of RID is compared against C-RIC and NC-RIC on the first group of the IMM Face database (i.e. the group where all subjects exhibit facial hair). The two RIC variants were trained on three levels of a Gaussian pyramid, on the same set of images as RID was, in order to reduce the effects of local minima on its generative fitting regime. To limit fitting time, C-RIC and NC-RIC were limited to 20 and 50 iterations per pyramid level, respectively. RID was limited to 20 appearance fitting steps in the first fitting iteration, where the appearance parameters are initialised to zero, and for the other iterations, appearance fitting was limited to 3 iterations. Note that due to the use of an appearance parameter mapping function, RID only requires a large number of appearance fitting steps in its first iteration, where initialisation is poor. For the other iterations, the mapping function provides a good initial estimate of the parameters, allowing convergence to local minima to be reached in far fewer steps.

Examining Plot (a) in Figure 5.12, it is clear that RID is not capable of achieving the same level of performance as its non-robust counterparts, discussed in Sections 5.2 and 5.3. It achieves only 90.77% convergence with an average accuracy of converged samples of 5.78

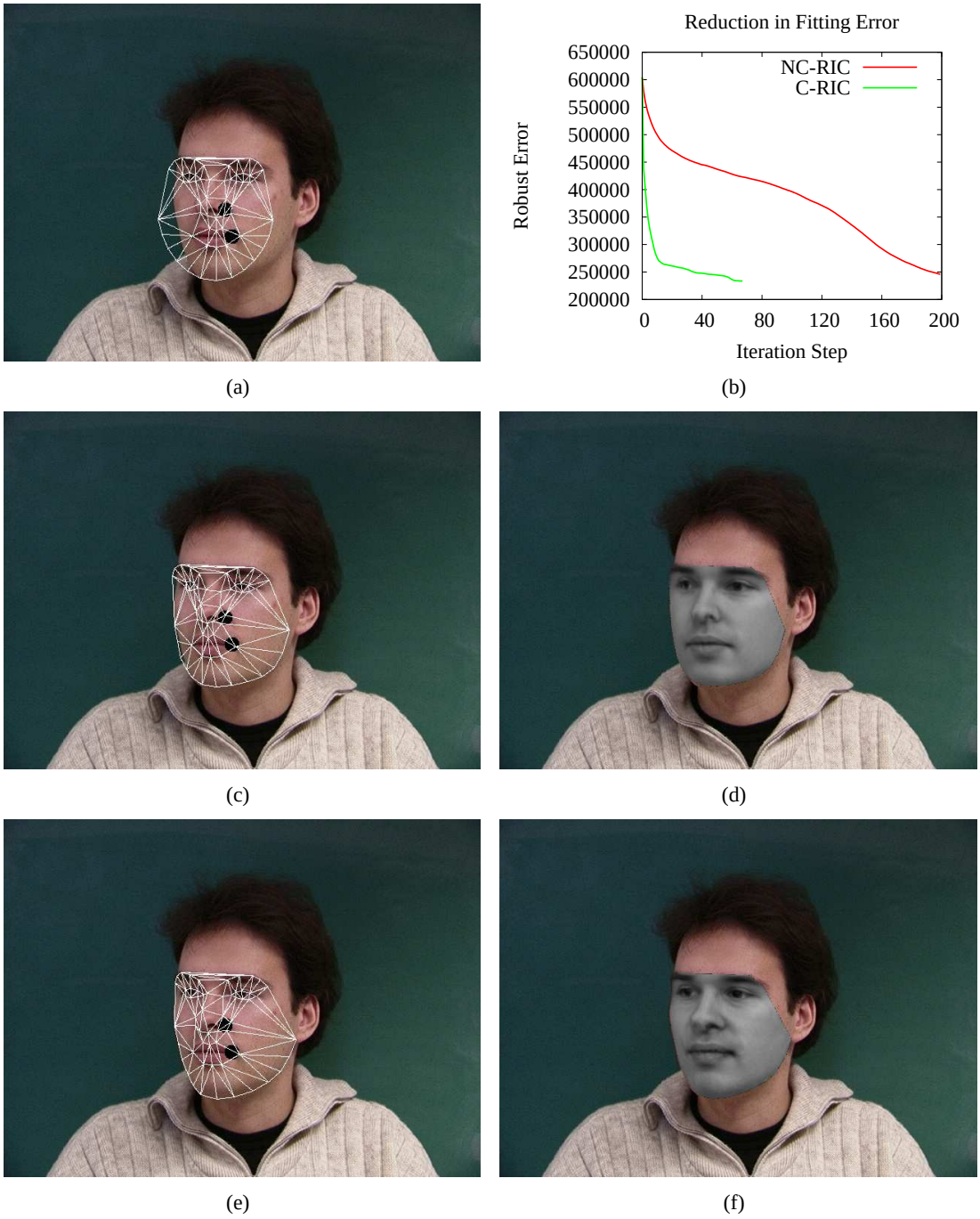


Figure 5.11: The effects of assuming spatially coherent occlusions in RIC. **(a):** initial setting from which the AAM is fitted using RIC. **(b):** the evolution of robust fitting error of for RIC using commensurate (C-RIC) and non-commensurate (NC-RIC) robust weights. **(c) and (d):** the AAM's shape and appearance at convergence using C-RIC. **(e) and (f):** the AAM's shape and appearance at convergence using NC-RIC.

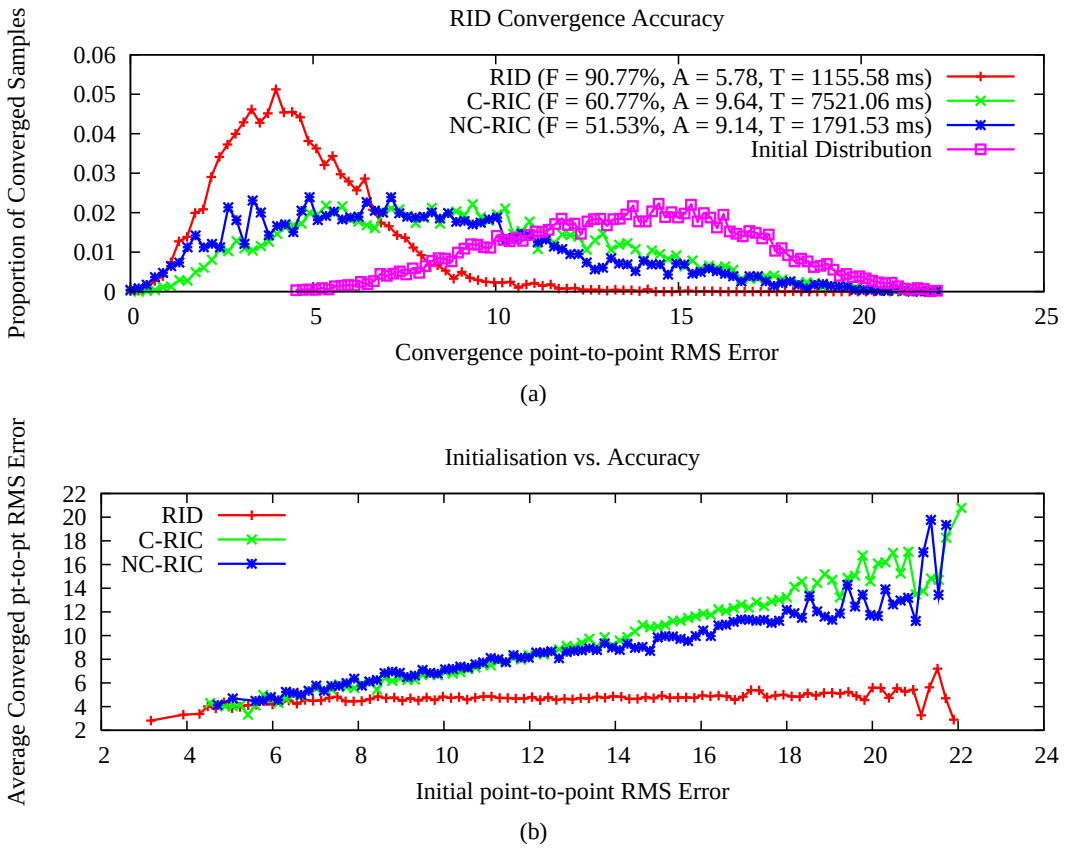


Figure 5.12: Performance comparisons between the RID, C-RIC and NC-RIC. **(a):** convergence accuracy histograms. **(b):** effects of initialisation on convergence accuracy.

point-to-point pixel RMS. Furthermore, processing time is significantly slower, requiring on average, 1155.58ms to fit an image. Nonetheless, compared to RIC it exhibits superior performance in all respects. The poor performance of both variants of RIC can be attributed to their evaluation on a generic person database. Previous results regarding its performance, reported in [55; 119], utilise only models with very limited variabilities. Since RIC is essentially a robustification of NIC (see Section 5.1), it can be expected to perform in a similar way to NIC. In fact, comparing the results for RIC in this section with that of NIC in Section 5.1, one notices the similarities in convergence rates, average convergence accuracy and even the convergence accuracy histograms. It should be noted though that RIC does not perform as well as NIC, possibly due to the approximations made regarding spatial coherence of the outliers. Out of the two RIC variants, C-RIC exhibits better average converged accuracy but poorer generalisability than NC-RIC. The better accuracy of C-RIC may be attributed to the fact that difficult samples do not converge with C-RIC, excluding their effects on the computed average converged accuracy. Examining Plot (b) in Figure 5.12, the utility of the ID approach is again evident in the consistently good average convergence accuracy of RID as initialisation becomes poorer, up to around 20 pixels point-to-point RMS error. As expected, RIC performs poorly in this respect, where, as with FJ, POIC, SIC and NIC, its average convergence accuracy deterio-

Table 5.2: Summary of the Synthetically Occluded Results

	5% Occlusion		10% Occlusion		20% Occlusion	
	A (RMS)	F (%)	A (RMS)	F (%)	A (RMS)	F (%)
Black	4.43	92.40	4.02	96.39	5.11	85.25
White	4.29	91.67	4.77	87.03	6.65	65.54
Random	4.41	92.72	4.34	93.03	4.72	90.73

A = Average converged accuracy F = Convergence rate

rates the further initialisation is from the true settings of AAM parameters. Finally, unlike the results reported in [55], where RIC affords real time fitting, here, the average fitting time is in excess of one second for C-RIC and seven seconds for NC-RIC. Again, this can be attributed to the large variabilities exhibited by the model, both in shape and appearance, which lead to an expensive parameter update procedure, which involves two inversions of Hessian sized matrices at each iteration.

In Table 5.2, results of the experiments on the synthetically occluded images are summarised. Although experiments where the occlusions are black and values are randomly selected give mixed results, the white occluded images show a strong trend of deterioration as the amount of occlusion is increased. Also, the performance on the white occluded images is inferior to both the black and randomly occluded images, both in convergence rate and accuracy, over all settings of occlusion percentage. It seems, therefore, that the robust feature fitting procedure is more sensitive to occlusions with high intensities. One explanation for this is that the robust feature extraction procedure is more prone to terminating in local minima with this type of occlusion.

In conclusion, although the robust variant of the inverse-compositional method that utilises a linear kernel fails to achieve the same level of performance as its non-robust variants, it still provides a significant step forward over the robust inverse compositional method on a generic face database. Through the utility of appearance parameter mapping, and assuming spatially coherent outliers, the method is capable of fitting an image with outliers in around one second. Although better performance may be achieved by utilising nonlinear kernels in the regressors, processing times will also increase due to the complexity in evaluating nonlinear kernels. Furthermore, the problem of regularisation is complicated in this case, since not only must the regularisation parameter be set, but also the parameter pertaining to the nonlinear kernel.

5.5 Background Invariant Fitting

The final variant of ID, which is proposed in Chapter 4.5, is that which accounts for background variabilities through a feature selection procedure at each iteration. Here, a pixel of the observed image is selected for inclusion in the feature vector, if it is identified as a foreground pixel (i.e. that pertaining to the visual object of interest in an image) in a large proportion of the training samples. Apart from this, all other elements of the fitting procedure follow that of the basic ID approach. Although any of the variants described previously can be used as a prototype, only the COTLID prototype is evaluated here because it performed the best in the

(a) (b)

Figure 5.13: Chrominance based background segmentation. **(a):** training data for background classifier. **(b):** examples of background segmentation.

previous experiments, deferring evaluation of the other prototypes as future work.

In order to segment background from foreground in the IMM Face database images, a non-parametric chrominance based background classifier is built from examples of foreground and background pixels. The training data used for this classifier is shown in Plot (a) of Figure 5.13, which consists of image patches from the database, selected manually. The classifier consists of two 2D-histograms of the background and foreground pixels in CIE-Lab's chrominance space, smoothed appropriately. The label of a pixel is then assigned as background or foreground depending on which of the histograms has a larger value at coordinates describing the pixel of interest, in chrominance space. Some examples of the results of using this background classifier are presented in Plot (b) of Figure 5.13. Since a small number of images in the IMM Face database are in greyscale, the background classifier cannot be used on these images. For this reason, for all experiments in this section, those images are removed from the experiment's image set. In the 4-fold cross validation experiments, the remaining image set is partitioned into four equally sized groups in such a way that each subject occurs in only one of these groups.

In training the background invariant iterative-discriminative method (BIID), at each iteration samples are obtained, both from the original image as well as the background masking image. Before learning the regressors for each parameter, the utility of each cropped pixel for inclusion in the feature vector to be used for regression is evaluated based on how many times it is identified as foreground over the whole sample set (in these experiments, as with those in previous sections, a sample size of $N = 2000$ is used at each iteration). A threshold for pixel inclusion was set at 99%, where pixels labelled as background in more than 1% over the sample set are removed from the feature vector. The choice for this threshold may appear somewhat arbitrary. However, if it is set too large, not enough pixels will be retained, resulting in insufficient data to perform accurate linear regression. On the other hand, if chosen too small, then an unacceptable number of components in the feature vector may correspond to background pixels during fitting, leading to unpredictable fitting behaviour. Training BIID with $\lambda = 10$, which has been shown in Section 5.2 to give good results on COTLID and

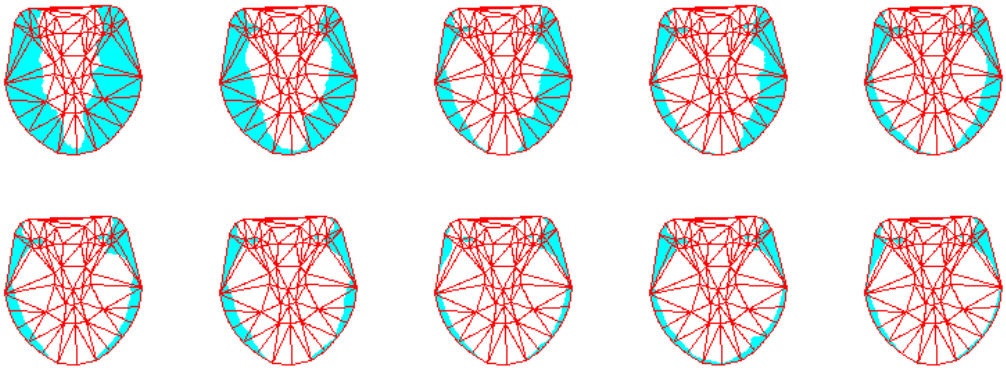


Figure 5.14: Left to right, row-wise: the evolution of features chosen for inclusion throughout the fitting procedure of BIID. Aqua coloured pixels denote those excluded from regression.

ATLID, the evolution of the features chosen for inclusion throughout the fitting procedure is illustrated in Figure 5.14. Notice that the region influenced by the background initially resides over large areas around the periphery of the canonical shape, but reduces in size as iterations progress and samples become more constrained around the optimum parameter settings.

To investigate the effects of varying background on BIID, three sets of experiments were performed using the 4-fold cross validation technique. In each, the background of the test images was set either to black (a pixel value of zero), white (a pixel value of 255), or a randomly sampled value within the range $[0, 255]$. The combined results of each of the three experiments is shown in Figure 5.15. Examining Plot (a), one immediately notices that the convergence rate over experiments with white background images is significantly poorer than experiments on the black or random background images, affording only 90.27% convergence. This significant difference can be attributed to the fact that during the fitting procedure, some samples will be perturbed into configurations where some background pixels will be included in the feature vector used for regression. The fact that this occurs, despite the procedure of feature exclusion, can be explained by three factors:

- Not all pixels were excluded from the feature vector that were labelled as background in some of the samples (i.e. only 99% of them).
- Even if an exclusion rate of 100% is used, this may still occur, due to the finite sample set used for excluding background pixels from the feature.
- Since performance on unseen images is expected to be poorer than that on the training set, some test instances will be updated into these undesirable configurations.

The effects of this on the black and random background is not as dramatic since the magnitude by which they perturb an update is smaller. Notice that the performance on the random background images is slightly poorer than on the black background images. Examining Plot (b), it seems that the effects of background variability is more pronounced with poor initialisation, as can be seen from the deterioration of the performance on the white background images,

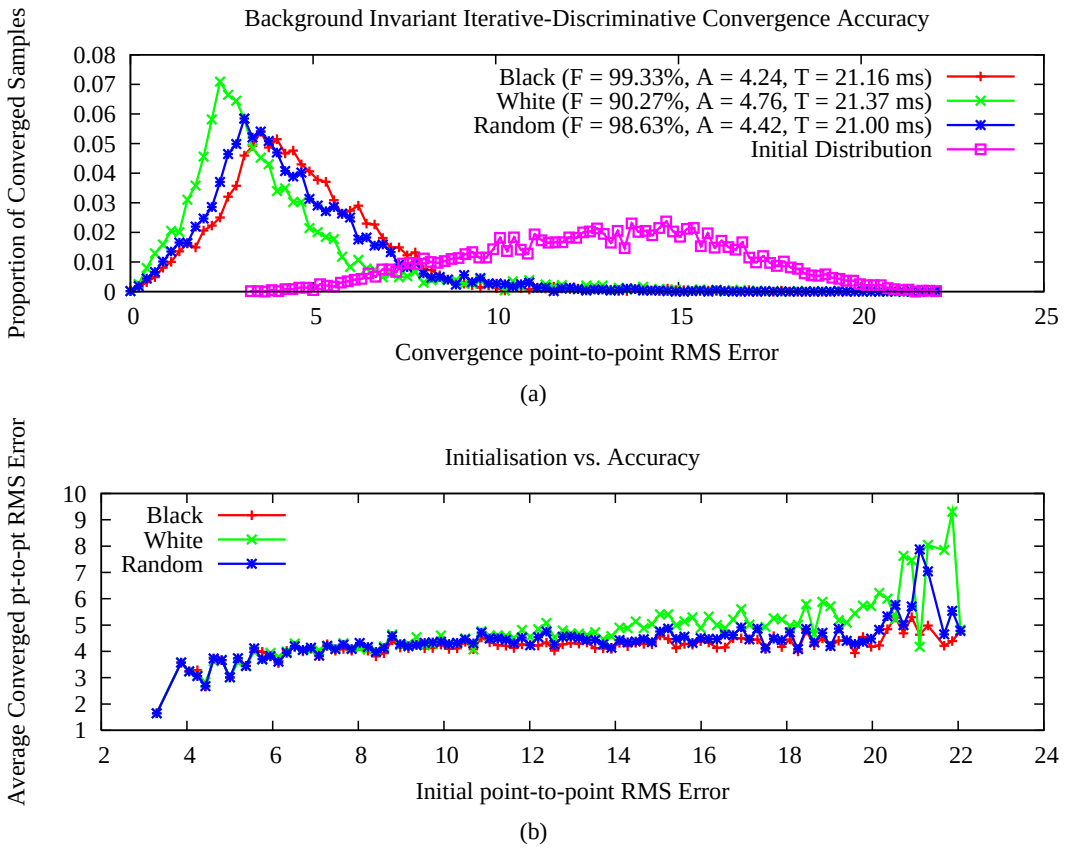


Figure 5.15: Performance evaluation of BIID on three different backgrounds: black, white and random. **(a):** convergence accuracy histograms. **(b):** effects of initialisation on convergence.

compared to the black and random ones, as initialisation error is increased above 12 pixels point-to-point RMS. Finally, it should be noted here that the accuracy of converged samples in the white background image seems to be better than the black and random background images. This performance difference is superficial, however, since the difficult samples, which diverge on images with a white background, are excluded from the computation of the average accuracy. In the cases with black and random background, these difficult samples still converge. However, their accuracy suffers due to the perturbation caused by the inclusion of background pixels in the estimation of their updates.

In conclusion, BIID has been shown to effectively handle background variabilities, especially in the case where the variations exhibit low intensities. Although performance deteriorates when the background pixels have very high intensities, the performance of BIID is still superior to the baseline methods discussed in Section 5.1, in all respects.

5.6 Conclusion

In this chapter, the efficacy of ID has been evaluated in the context of generic face model fitting. Three aspects of the problem were assessed here: performance in an outlier free setting, ro-

bustness towards outliers, and invariance to background variability. For each of these, separate ID prototypes were evaluated, each specialised to the aspect of the problem being assessed. As a result, the conclusion may be drawn that ID is a powerful technique that excels in dealing with the problem of generic face model fitting.

In the outlier free case, three prototypes were evaluated: ATLID, COTLID and HFBID. All three were shown to outperform four existing baseline methods in convergence accuracy. Most notably, however, was their significant improvement in convergence rate, attaining convergence in almost every trial. Furthermore, the computational complexity of these prototypes is much smaller than the baseline methods, even compared to POIC, which is perhaps the most efficient AAM fitting procedure to date. Compared to HFBID, ATLID and COTLID exhibited marginally better performance. However, this may be attributed to the small sample size of features used in the HFBID's boosting procedure.

In the case of images with outliers, such as those caused by unmodelled appearance or occluding objects, the efficacy of the RID prototype was assessed. Through extensive experiments, both over synthetically occluded images as well as real unmodelled appearance variations, it was shown that RID significantly outperforms RIC in all aspects. However, due to its robust feature extraction procedure, which involves a generative appearance fitting process, the fitting times afforded by RID are much slower than its non-robust counterparts. Nonetheless, on the generic face fitting problem, where the model typically exhibits a large number of modes of variation, RID is still more efficient than RIC.

Finally, the ID prototype BIID was used to assess the ability to handle background variabilities. It was found that when the background variations are well behaved (i.e. they exhibit small intensities), then BIID has the potential to approach the performance of ATLID and COTLID. However, when the background exhibits high intensities, the generalisability of BIID suffers. Nonetheless, compared to the non-robust baseline methods, BIID still exhibits better performance, even with a white background. Furthermore, BIID affords extremely rapid fitting, even more so than ATLID and COTLID, since the number of features in its regressors is comparatively reduced.

There are a number of avenues for future work on the ID prototypes that can be pursued. One of the most straightforward is to investigate the effects of choosing better parameters to use in the various discriminative learning problems. This might include a more elaborate selection scheme for the regularisation parameter λ, the tuning of various sample set sizes, such as N, N_d and N_c, or the choice regarding non-linear kernels for use in a non-linear regressor. Another task for further development is to evaluate the effects of combining the features of the various prototypes. Although separate prototypes were utilised in this chapter, in order to assess the various aspects of generic face model fitting, it should be possible to combine the various prototypes into one. For example, COTLID can be combined with RID and BIID to obtain a robust background invariant prototype that trains its linear regressors using the ν-SVR method. Finally, the utility of ID would benefit from addressing its main drawback: its extended training time.

Conclusion

Black then white are all I see in my infancy.
Red and yellow then came to be, reaching out to me,
lets me see.

Tool

The aim of this study has been to investigate the utility of the Linear Deformable Model (LDM) for the task of modelling deformable visual objects, as well as the automatic extraction of their structure from images. The main drawbacks of the LDM, namely the difficulty of automatic data collection and the opposing criteria of efficient, accurate and reliable structure recovery (fitting), are addressed, at least in part, through a principled treatment of these problems. The novel solutions proposed in this thesis are empirically evaluated through extensive experimentation on the challenging task of human face modelling and fitting.

The problem of data collection arises due to the LDM's parameterisation that uses statistical models of shape and appearance to represent a deformable visual object. These statistical models are simultaneously LDM's greatest strength and weakness. They constitute a strong global prior on the space of allowable variations, allowing only valid instantiations of the visual object to be generated. However, a large number of correspondences across a training set of images is required in order to facilitate the learning of these statistical models of shape and appearance. Manual selection of correspondences is both tedious and error prone, biased by the subjectivity of the human expert. Furthermore, for some flavours of the LDMs, such as the 3DMM, manual annotation is not tractable since a dense correspondence set is required. The automatic extraction of correspondences, on the other hand, also presents significant difficulties. Inherent differences in appearance between various instantiations of the same deformable visual object means that correspondences found through photometric criteria alone are often misleading. As such, geometric constraints must be deployed to complement inference from photometric constraints. However, unlike the problem of rigid object correspondence, the shape variations inherent in deformable visual objects present a significant challenge since well established geometrical relations between images are not applicable. The resulting problem is therefore underconstrained, requiring the incorporation of domain knowledge into the formulation in order to attain meaningful correspondences. How the problem should be posed in order to make the best use of domain knowledge, however, is nontrivial.

The recovery of a visual object's structure from an image is often tackled by an LDM through an analysis-by-synthesis procedure. Stemming from its generative construction through the utility of statistical models of shape and appearance, fitting is posed as a nonlinear optimisation problem, whereby the LDM's parameters are iteratively perturbed such that its appearance best matches the image region defined by the LDM's shape. The difficulty lies in computing the parameter updates, a process that generally requires the computation of the cost function's

gradient. For many visual objects, this is a computationally expensive procedure, leading to an inefficient fitting procedure. Although there exist numerous approximations that are capable of reducing the computational burden involved, they tend to deteriorate the fitting accuracy due to a mismatch between the true objective and its approximation. Some reformulations have also been proposed such that portions of the computation can be moved to the training phase. However, some of these reformulations fail to reduce the computational cost sufficiently, while others can accommodate only simple visual objects that exhibit small amounts of variations. Another difficulty is a product of the generative formulation that is deployed. Since the relationship between the LDM parameters and the image's pixel values is generally nonlinear, the cost function often exhibits many local minima, in which the fitting procedure can potentially terminate. Although the effects of local minima can be reduced through the utility of multiple features as well as optimising on a Gaussian pyramid, this difficulty has not yet been addressed in its entirety. Finally, there also exist problems with model generalisability, where the training and test images are mismatched, leading to unreliable fitting.

6.1 Summary of Contributions

This dissertation makes contributions towards solving each of the drawbacks of LDMs described above. A summary of these contributions is outlined for each of these below.

6.1.1 The Pairwise Learning of Correspondences

In Chapter 3, the problem of deformable visual object correspondence was tackled from a direct pairwise perspective. There, pseudo-dense correspondences between a template and all other images was pursued by deforming the template, both in shape and appearance, such that it best matched the other images. A formal treatment was afforded by formulating the problem within a Bayesian Framework. It was shown that the popular regularised data fitting problem, often deployed in existing nonrigid correspondence learning methods, can be derived directly from the proposed formulation. Using the method of hierarchical priors, complemented by the *marginalised maximum likelihood/maximum a posteriori* (MML/MAP) iterative optimisation scheme, the proposed approach affords the automatic tuning of all free parameters in the problem. In particular, weightings between the data and regularisation terms in the regularised data fitting problem, which correspond to the hyperparameters of the Bayesian formulation, were optimised in conjunction with the correspondences. In most existing works, these weights are often chosen manually, requiring a trial and error procedure to find their best setting. Finally, the EM procedure was deployed for the task of optimising the hyperparameters, affording simple forms for their updates that guarantee that at least a local optimum is reached, without resorting to general purpose numerical optimisation techniques that generally require the manual selection of various optimisation parameters, such as the step size.

To instantiate this general approach for correspondence learning, specific instances of the image likelihood and priors over deformations were proposed. Geometric constraints were placed on the deformable template through the assumption that deformations exhibit piecewise smoothness. This was achieved by penalising differences between deformations of adjacent landmarks in the object's shape using an M-estimator, weighted by their proximity in the

template. This is a realisation of the smoothness assumption, which implies a level of topological rigidity. Priors over deformations were then modelled as Gibbs priors, with the weighted sum of neighbouring deformation differences constituting the Gibbs energy. To account for differences in appearance between various instantiations of the deformable visual object, the template's appearance was allowed to deform along with its shape. To restrict the space of possible variations, the appearance model was constrained to represent appearance differences that vary slowly over the template. In this case, an extension of the piecewise affine warp was proposed that is capable of modelling piecewise affine appearance differences between the template and the image. To account for large appearance differences that are spatially localised and unaccounted for by the appearance model, the matching residual was embedded within an M-estimator. The image likelihood, then consisted of the robust error measure between the template and the image in its energy term. To address the difficulty of evaluating the integral in the MML problem, the image likelihood and deformation priors were approximated by Gaussian PDFs, which resulted in the joint likelihood of the complete data (i.e. shape and image) also taking the form of a Gaussian PDF. This form affords an analytic solution to its improper integral. The approximation was attained by applying a first order Taylor expansion over the image and all robust estimators in the formulation. A full derivation of this approximation was presented, allowing adaptations to similar problems to be made easily.

Empirical Evaluation

The efficacy of the proposed pairwise method was empirically evaluated on three types of datasets: a person specific, pose specific and a generic person dataset. The person specific dataset included variations in pose, expression and lighting. The pose specific dataset included variations in identity with fixed pose, lighting and expression. Finally, the generic person dataset consisted of a combination of the sources of variation in the other two datasets. Two sets of six experiments were conducted on these datasets. The first set of experiments were designed to evaluate the modelling capacity of the proposed pairwise method by starting the optimisation from manually annotated correspondences. In the second set of experiments, the sensitivity of the proposed method to local minima was investigated by using a bounding box detected initialisation. In each set, the six experiments constitute different combinations of robustifications of the likelihood and priors. From the six conducted experiments in the first set, it was found that, in a person specific case, the effects of assuming piecewise smoothness as compared to strictly smooth deformations had little effect. In the pose specific dataset, the effect of this choice was more pronounced, with a slight improvement in accuracy observed by utilising robust deformation priors. In contrast, the effects of utilising the piecewise smooth assumption in the generic person dataset was quite marked, as significant deterioration in performance was observed when only strictly smooth deformations were allowed. From the same set of experiments, the utility of the appearance deformation was also demonstrated. As with the piecewise smooth assumptions on spatial deformation, the effects of allowing the template's appearance to deform along with its shape became more pronounced as the complexity of the dataset increased from the person specific to the pose specific to generic person datasets. The conclusion can therefore be drawn that in the pairwise setting, modelling deformations as piecewise smooth and allowing the appearance to deform, also in a piecewise smooth fashion,

exhibits the best overall modelling capacity. However, the results from the second set of experiments suggest that the good modelling capacity of the proposed method comes at a the cost of sensitivity to local minima. For this reason, in order to attain meaningful correspondences, the proposed pairwise method should be used in conjunction with either a more sophisticated initialisation mechanism than a bounding box or extra features to smooth the objective function. The proposed pairwise approach presents itself readily for adaptations to other similar problems. An example of this was presented in Appendix B, where a full derivation of the adaptation of the proposed approach to the problem of groupwise correspondence learning was presented.

6.1.2 Iterative-Discriminative Fitting

In Chapter 4, a novel approach to LDM fitting was proposed: the iterative-discriminative approach. It leverages on the predictive capacity of discriminative methods coupled with the iterative framework of generative fitting. Utilising the error-bound minimisation paradigm, a continuity in objective is enforced between the iterations, guiding samples at all locations towards their respective optimum, placing a higher priority on those that are poorly predicted. The approach promotes the realisation of a fitting procedure that achieves the best *overall* performance rather than *specific* instances of the visual object. The utility of error-bound minimisation also has the effect that the objective at each iteration only needs to be partially satisfied. This allows simple regression functions to be utilised as predictors at each iteration. This in turn leads to a rapid fitting procedure and better generalisation, since simple predictors exhibit better generalisation properties than more complex ones. The proposed training procedure also promotes further generalisability through a resampling process that artificially increases the training set size without significantly increasing training time. Finally, it has been shown that the proposed method is highly applicable, with no specific requirements placed on the model's parameterisation or the types of features used to drive the predictions.

To realise the iterative-discriminative approach, a number of prototypes were proposed, each of which were designed to tackle different components of the LDM fitting problem. The first prototype was one that utilised a linear predictive model. Two training procedures were proposed for this prototype. In the first, a *soft* error bound minimisation was achieved though the utility of the linear ν-SVR's framework. This method enforces error bound minimisation over a large proportion of the samples, with the remaining ones captured through the use of slack variables. The second proposed training method enforces strict error bound minimisation over all training samples. For this, a new cost function was proposed for its training, namely the asymptotic penalty. This cost function asymptotically penalises errors on samples as a function of the parametric distance from their optimal settings. An optimisation strategy for this convex cost function was also presented, deriving the forms for its direction update and line search components.

The second proposed prototype of the iterative-discriminative approach was one that utilised a nonlinear regression function that consisted of a convex combination of a set of nonlinear weak learners, learnt through a boosting-like procedure. This prototype was proposed to account for cases where the capacity of a linear model is insufficient to afford accurate predictions. To attain high efficiency, a novel multimodal weak learner was proposed that uses the

Haar-like features to drive their predictions. A complete training procedure was outlined for this prototype that involved an adaptation of the asymptotic penaliser to the one dimensional case. To handle more general predictive models, a second nonlinear prototype of the iterative-discriminative method was proposed that utilises the nonlinear ν-SVR method for its training. In order to reduce computational complexities in fitting, this prototype utilised a dimensionally reduced feature, attained by applying PCA to a training set of raw features.

To account for fitting problems where the visual object exhibits occlusional effects or unmodelled appearance variations, the second nonlinear prototype was robustified at the feature extraction stage. This involved minimising the appearance difference between an occlusion ridden raw feature with that generated by a linear model, composed within a robust error measure, once for each iteration of the fitting procedure. Although the forms of the appearance parameter updates are much simpler than their counterpart in generative fitting problems, since a full optimisation must be performed at each iteration, the resulting method can still be computationally expensive. Two measures were takes to reduce the computational cost involved here. In the first, the previously proposed assumption regarding the spatial coherence of outliers was used to minimise the computational complexity of the Hessian in its Gauss-Newton optimisation scheme. The second measure used to reduce computational cost was the utility of a parameter mapping function that related appearance parameters between consecutive iterations of the whole fitting procedure. This mapping function allows reasonable initial estimates of the appearance parameters to be computed, allowing the Gauss-Newton optimisation to attain convergence in fewer iterations than if a random initialisation was utilised.

The last iterative-discriminative prototype proposed in Chapter 4 was one that accounted for background variability. This was achieved by excluding those features that were labelled as part of the background in a preselected small fraction of training samples from the features passed to the regressors. This method relied on the assumption that a reasonable initialisation of the model is available, in which case background effected features reside on the periphery of the visual object only. As fitting iterations proceed and estimates of all samples improve through the satisfaction of error bound minimisation, the background affected region around the periphery of the object reduces in size, allowing more features to be used to attain better predictions.

Experimental Evaluation

In Chapter 5, the various prototypes proposed in Chapter 4 were empirically evaluated on the difficult problem of generic face fitting. The AAM parameterisation was utilised in all experiments, where in order to provide a relative scale of performance, five existing baseline methods for AAM fitting were also implemented. Through comparisons of their respective 4-fold cross validation experiments, the two linear prototypes of the iterative-discriminative approach were shown to significantly outperform all other non-robust baseline methods in overall convergence accuracy and reliability. Furthermore, this significant improvement was attained whilst affording an extremely rapid fitting time, even more so than the project-out inverse compositional method, which is commonly considered the fastest LDM fitting method. Although the Haar-based nonlinear prototype also exhibited significant performance improvements over the baseline methods, its performance was not as impressive as its linear counterparts. This can be

explained by the fairly coarse selection of weak learners and parameters used in the boosting procedure, which was required due to time constraints placed on this study. However, a better selection and a larger number of weak learners can be expected to improve on the results reported here.

The robust extension of the nonlinear prototype was compared against a prominent robust generative fitting procedure. As a complementing contribution, an investigation was made into the choice regarding commensurate and non-commensurate robust weights used in computation of the updates of the robust generative method. It was found that the use of commensurate scalings gives an improvement in performance of this method. However, compared to the robust iterative-discriminative prototype, it was significantly outperformed in convergence accuracy, reliability and fitting time. An investigation into the effects of outlying pixel values was also performed on the robust iterative-discriminative prototype. It was found that outliers with larger values affected the fitting procedure more severely than smaller valued outliers.

Finally, the efficacy of the background invariant method was evaluated on a partition of the database for which background segmentation could be achieved automatically, where the affects of different background values on this prototype's performance was investigated. As with the other prototypes, this method was shown to significantly outperform the baseline methods in all respect. Compared to the linear iterative-discriminative prototypes, it failed to achieve the same level of accuracy and reliability, although it did exhibit smaller fitting times. The efficacy of the method deteriorated however, when background pixels exhibited very large values.

6.2 Future Work

The contributions presented in this dissertation constitute general frameworks within which a number of extensions and further experimental evaluations can be performed. Some directions for future work for both major areas of contribution are outlined below.

Automatic Model Building

By virtue of its Bayesian framework, the formulation of the proposed approach for pairwise correspondence learning allows a number of extensions to be pursued:

- *Additional Features*: The high sensitivity of the pairwise method to local minima is a product of the highly nonlinear cost function that it attempts to optimise. It has been shown previously in [101] that the addition of a number of different features into an objective function has the effect of *smoothing out* fluctuations in the cost function, reducing the likelihood of an optimisation terminating in local minima. The same approach can be applied here, where features take the form of multiple prior and likelihood terms. This may include an image gradient based likelihood, as utilised in variational optical flow [22; 110]. Another example is to use a small set of manually annotated correspondences to constrain fitting [101]. Since there are a large number of such features that can be included into the framework, a trial and error procedure must be utilised here, in order to find the best set of features for a given problem.

- *Reparameterisation*: The approach proposed here is not dependent on the parameterisations of the likelihood and priors. Although reasons for the choices made here are given at the time of their introduction, they may be suboptimal. Examples of this include the type of M-estimator used, both in the likelihood and prior, the kernel used as a weighting function in the prior, and the parameterisation of the warp and appearance generating function. Modifications of any of these components does not affect the general derivation of the approach proposed here, only the particular instances of the equations.

- *Groupwise Extension*: As mentioned earlier, an adaptation of the pairwise approach proposed here has been fully derived in Appendix B for the problem of groupwise correspondence learning. This method can potentially provide better modelling capacity than the pairwise method, since it uses a linear model to represent shape and appearance, composed with extrinsic normalising functions, which has been shown to perform well in the related problem of LDM fitting. Furthermore, its approximation of the joint likelihood required by the MML procedure may be better since it requires only a first order Taylor expansion of the cropped image, rather than of the robust functions as well. An obvious next step in future work, therefore, would be to implement and evaluate its efficacy.

Accurate, Reliable and Efficient Fitting

The proposed iterative discriminative approach forms a framework within which various modifications can be made to improve the performance of LDM fitting. Amongst others, directions of future work may include:

- *Optimal Parameters*: Although extensive experiments comparing the performance of the various prototypes of the iterative-discriminative approach with some prominent fitting methods have been performed in this study, less rigour has been applied on the task of choosing the optimal parameters with which to train the methods. These parameters include the various regularisation parameters, the number of features and various other thresholds, such as the inclusion rate of the background invariant method. Since discriminative methods can be quite sensitive to the choice of these parameters, further improvements can be expected of these methods when the optimal parameters are chosen. However, this requires a lengthy cross validation procedure[1].

- *Optimal Regressors*: Although an example of an implementation using a nonlinear regressor has been presented, there are a large number of nonlinear predictors that can be utilised here. It is a straight forward process to apply different types of nonlinear kernels to the nonlinear prototype proposed here. In fact, using `DeMoLib` that has been provided with this dissertation in the enclosed CD-ROM, experiments on this aspect can be readily performed, requiring only a significant processing time for their training.

- *Feature Combination*: The various prototypes of the iterative-discriminative approach address a particular aspect of the problem of LDM fitting. There is no reason why these

[1] The training procedure of each of the prototypes of the iterative-discriminative approach presented in Chapter 5 took around eight to ten hours each, on a 3GHz Pentium 4 machine.

prototypes cannot be combined into one. For example, combining the robust and background invariant method may result in a prototype that is capable of handling occlusions without being affected by background variabilities. Also, the iterative-discriminative method proposed here can be combined with a slow but accurate generative fitting procedure to further improve fitting performance, where the iterative-discriminative method provides an excellent initialisation to the generative method such that it can avoid local minima and afford faster convergence.

The Extended Piecewise Affine Warp

The piecewise affine warp, commonly used in AAM representations [30; 83], is a spatial transformation function for which the *reference frame* is divided into a set of non-overlapping regions such that all locations within each region are warped using the same affine transformation. These regions are generally defined by some type of triangulation of a point set, such as the Delaunay triangulation [36], defined in the reference frame. The result is that locations in the reference frame are warped to locations in the *destination frame* with the same *barycentric* coordinates, with respect to its encompassing triangle. It should be noted that the piecewise affine warp is defined only within the convex hull of the point set defining the triangulation.

Consider a set of 2D points in the reference frame $\{\mathbf{x}_i\}_{i=1}^n$. For a given set of 2D points in the destination frame $\{\vec{\mathbf{x}}_i\}_{i=1}^n$, the piecewise affine warp is defined as follows:

$$\mathscr{W}\left(\mathbf{x}; \{\mathbf{x}_i\}_{i=1}^n, \{\vec{\mathbf{x}}_i\}_{i=1}^n\right) : \Re^{2n} \rightarrow \Re^2 = \vec{\mathbf{x}}_i + \alpha_{\mathbf{x}}(\vec{\mathbf{x}}_j - \vec{\mathbf{x}}_i) + \beta_{\mathbf{x}}(\vec{\mathbf{x}}_k - \vec{\mathbf{x}}_i), \tag{A.1}$$

where $\mathbf{x} \in \text{tri}\{\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k\}$. Here,

$$\alpha_{\mathbf{x}} = \frac{(\mathbf{x} - \mathbf{x}_i)^T \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} (\mathbf{x}_k - \mathbf{x}_i)}{(\mathbf{x}_j - \mathbf{x}_i)^T \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} (\mathbf{x}_k - \mathbf{x}_i)} \quad \text{and} \quad \beta_{\mathbf{x}} = \frac{(\mathbf{x} - \mathbf{x}_i)^T \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} (\mathbf{x}_j - \mathbf{x}_i)}{(\mathbf{x}_j - \mathbf{x}_i)^T \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} (\mathbf{x}_k - \mathbf{x}_i)} \tag{A.2}$$

are the barycentric coordinates of $\mathbf{x}$ with respect to the triangle with vertices $\{\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k\}$. Substituting Equation (A.2) into Equation (A.1), the typical affine form of the warp can be recovered:

$$\mathscr{W}(\mathbf{x}) = \mathbf{A}_{ijk} \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} \quad . \tag{A.3}$$

However, for the purpose of image registration, the form in Equation (A.1) is more useful since the reference frame and the points defining the triangulation are fixed, but the destination points are allowed to vary. As such, the barycentric coordinates of all the desired locations within the reference frame's valid domain, $\mathbf{x} \in \mathbf{\Omega}$, can be precomputed.

From the form in Equation (A.1), the observation can be made that for a fixed $\mathbf{x}$ in the reference frame, the x coordinate of the warped point is parameterised only by the x coordinates of the destination triangle $\{\vec{x}_i, \vec{x}_j, \vec{x}_k\}$, and similarly for the y coordinates. In fact, for a fixed

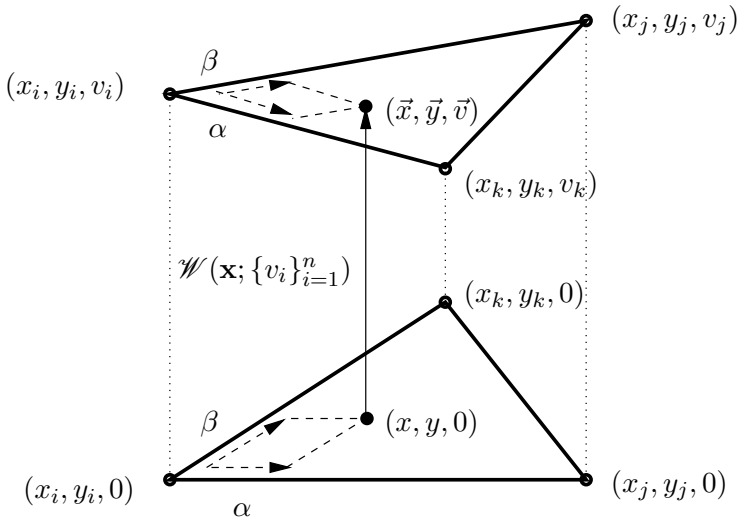


Figure A.1: Illustration of the 1D piecewise affine warp.

reference frame, the warp defines a simple linear interpolation on the values of the destination vertices, separately for each dimension. As such, the piecewise affine warp can be generalised to constitute a piecewise linear interpolation of any dimension. In Figure A.1, this interpolation is illustrated for the 1-dimensional case, mathematically represented as:

$$\mathscr{W}(\mathbf{x}; \{\mathbf{x}_i\}_{i=1}^n, \{\vec{v}_i\}_{i=1}^n) : \Re^n \rightarrow \Re = \begin{bmatrix} 1 - \alpha_{\mathbf{x}} - \beta_{\mathbf{x}} & \alpha_{\mathbf{x}} & \beta_{\mathbf{x}} \end{bmatrix} \begin{bmatrix} \vec{v}_i \\ \vec{v}_j \\ \vec{v}_k \end{bmatrix} \quad . \tag{A.4}$$

With this representation, the piecewise affine warp is extended from a purely spatial transformation function to a transformation of any quantity, where the linear interpolation is defined by the fixed 2D points and valid locations in the reference frame. For example, in Chapter 3, this formulation is used to represent piecewise linear appearance transformations.

Finally, we note that for a fixed reference frame, the derivative of the extended piecewise affine warp is constant, given by the following expression:

$$\nabla_{\vec{v}_l} \mathscr{W}(\mathbf{x} \in \mathbf{\Omega}_{ijk}) = \begin{cases} 1 - \alpha_{\mathbf{x}} - \beta_{\mathbf{x}} & \text{if } l = i \\ \alpha_{\mathbf{x}} & \text{if } l = j \\ \beta_{\mathbf{x}} & \text{if } l = k \\ 0 & \text{otherwise} \end{cases} , \tag{A.5}$$

where $\mathbf{\Omega}_{ijk}$ denotes the reference frame locations within $\text{tri}\{\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k\}$ for which the barycentric coordinates can been precomputed.

The Groupwise Learning of Correspondences

The pairwise method for automatic correspondence learning described in Chapter 3 exhibits good flexibility for modelling a large class of deformable visual objects, relying only on the assumption that spatial and appearance deformations are (piecewise) smooth. However, for some visual objects, this assumption may be too flexible as can bee seen from the results in Section 3.6, where the method is highly sensitive to local minima. Furthermore, it is shown that the pairwise approach is not suitable for visual objects exhibiting large amounts of shape and appearance variations. When prior knowledge about the specific type of distributions describing the shape and appearance of a visual object is available, better results may be achieved by optimising the parameters of these distributions directly, since the optimisation procedure becomes more constrained.

In the following, an adaptation of the procedure outlined in Chapter 3 to the case of groupwise learning of correspondences is presented, where correspondences across all images are learnt simultaneously. A formulation of the Bayesian framework for groupwise correspondence learning is presented in Section B.1. The objective of MML estimation is then discussed in Section B.2. The Expectation Maximisation (EM) algorithm used to solve the MML problem is then presented in Section B.3.

B.1 Dependence, Densities and Parameterisation

In a groupwise setting, the MAP formulation for automatic correspondence learning can be written as follows:

$$p\left(\{\mathbf{s}_i\}_{i=1}^N, \boldsymbol{\theta} | \{\mathscr{I}_i\}_{i=1}^N\right) = \prod_{i=1}^N p\left(\mathbf{s}_i, \boldsymbol{\theta} | \mathscr{I}_i\right), \tag{B.1}$$

assuming independence between shapes in every image. Here, $\boldsymbol{\theta}$ denotes the parameters that describe the visual object's distribution. Compared to the pairwise formulation, which assumes a separate and independent parameterisation defining the distributions in each image, here the visual objects in all images are described using one model, restricting the model complexity over the whole dataset. However, this formulation requires optimisation over all images to be performed simultaneously, rather than separately, as in the pairwise case.

Assuming independence between the shape and appearance of the visual object of interest, the posterior of each image in the training set takes the form:

$$\begin{aligned} p(\mathbf{s}, \boldsymbol{\theta}|\mathscr{I}) &= \frac{p(\mathscr{I}, \mathbf{s}|\boldsymbol{\theta}_s, \boldsymbol{\theta}_t)\; p(\boldsymbol{\theta}_s, \boldsymbol{\theta}_t)}{p(\mathscr{I})} \\ &\propto p(\mathscr{I}, \mathbf{s}|\boldsymbol{\theta}_s, \boldsymbol{\theta}_t) \\ &\propto p(\mathscr{I}|\mathbf{s}, \boldsymbol{\theta}_t)\; p(\mathbf{s}|\boldsymbol{\theta}_s), \end{aligned} \tag{B.2}$$

where the image index has been dropped for clarity of exposition and $\boldsymbol{\theta}$ in Equation (B.1) is decomposed into its components $\boldsymbol{\theta}_s$ and $\boldsymbol{\theta}_a$, pertaining to the shape and appearance, respectively. A non-informative prior is assumed for both $\boldsymbol{\theta}_s$ and $\boldsymbol{\theta}_t$. Here, $p(\mathscr{I}|\mathbf{s}, \boldsymbol{\theta}_t)$ denotes the image likelihood, which relates to the data term in the regularised data fitting framework. The prior over the corresponding shapes $p(\mathbf{s}|\boldsymbol{\theta}_s)$ relates to the regularisation term. Together, Equations (B.1) and (B.2) constitute a general Bayesian framework for groupwise correspondence learning, the realisation of which depends on the type of PDFs assumed for the likelihood and prior. The objective of the Bayesian inference in a groupwise setting, then, takes the form:

$$p\left(\{\mathbf{s}_i\}_{i=1}^N|\{\mathscr{I}_i\}_{i=1}^N\right) = \prod_{i=1}^{N} \int p(\mathscr{I}_i|\mathbf{s}_i, \boldsymbol{\theta}_t)\; p(\mathbf{s}_i|\boldsymbol{\theta}_s)\; d\boldsymbol{\theta}_s\; d\boldsymbol{\theta}_t. \tag{B.3}$$

In this dissertation, focus is primarily on visual objects that can be adequately described by an LDM. There is ample evidence in the literature, for example [11; 101], which suggests that for many visual objects that can be adequately be represented by an LDM, the distribution of both the object's shape and appearance approximates that of a multivariate Gaussian distribution[1]. However, directly modelling the full Gaussian distribution may be difficult to implement in practice, due to the large dimensionality of the visual object's appearance. Furthermore, since visual objects that can be adequately represented by LDMs have their data lie in a much smaller subspace than the dimensionality of the data, utilising a full multivariate Gaussian distribution here may lead to overfitting. As such, inspired by the work in [126] on non-rigid structure from motion, the shape $\mathbf{s}$ and appearance $\mathbf{t}$ of the visual object is modelled using probabilistic PCA (PPCA) [125]:

$$\mathbf{t} = \alpha\left(\bar{\mathbf{t}}^{(P)} + \boldsymbol{\Phi}_t^{(P\times M_t)}\mathbf{p}^{(M_t)}\right) + \beta\mathbf{1}^{(P)} + \boldsymbol{\epsilon}_t^{(P)} \tag{B.4}$$

$$\mathbf{s} = \mathbf{R}\left(\bar{\mathbf{s}}^{(2n)} + \boldsymbol{\Phi}_s^{(2n\times M_s)}\mathbf{q}^{(M_s)}\right) + \mathbf{T} + \boldsymbol{\epsilon}_s^{(2n)}, \tag{B.5}$$

where:

$$\mathbf{R} = \mathbf{I}^{(n\times n)} \otimes \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \quad \text{and} \quad \mathbf{T} = \mathbf{1}^{(n)} \otimes \begin{bmatrix} t_x \\ t_y \end{bmatrix}. \tag{B.6}$$

Following the convention set out in Chapter 3, here, N denotes the total number of training

[1] In fact, the Gaussian assumption on the distribution of shape and appearance of LDMs is implicit in their use of PCA that fits a Gaussian hyperellipsoid to the data. Although other dimensionality reduction methods can be used in LDMs (for example, the work in [131] uses Independent Component Analysis), by far the most common method used here is PCA. Since the use of this representation has been shown in many works to give good results, the use of Gaussian densities to model shape and appearance here is a reasonable approximation.

images, P denotes the number of pixels representing object appearance in the canonical frame, n denotes the number of landmarks in the shape, M_t denotes the number of appearance modes of variation, and M_s denotes the number of shape modes. In Equations (B.4) and (B.5), $\boldsymbol{\epsilon}_s$ and $\boldsymbol{\epsilon}_t$ are zero mean Gaussian noise vectors:

$$\boldsymbol{\epsilon}_t \sim \mathcal{N}\left(\mathbf{0}^{(P)}, \sigma_t^2 \mathbf{I}^{(P\times P)}\right) \quad \text{and} \quad \boldsymbol{\epsilon}_s \sim \mathcal{N}\left(\mathbf{0}^{(2n)}, \sigma_s^2 \mathbf{I}^{(2n\times 2n)}\right). \tag{B.7}$$

From the formulation above, it should be noted that the linear intrinsic models for both shape and appearance are composed with global transformation functions, which account for extrinsic sources of variation. The use of these transformations, here, is required since, although the multivariate Gaussian distribution can adequately model intrinsic variations of linear visual objects, this assumption is not well justified when external factors are involved. For example, modelling an in-plane rotation as a linear combination of bases does not lead to a Gaussian distribution over the shape, where certain combinations of the linear bases can generate implausible shapes. Even if the extrinsic sources of variation can be modelled accurately by a set of linear bases, generating little or no implausible instances, the distribution of the driving parameters cannot be assumed to be Gaussian. Examples of this are the shape's translation and scale, which are more accurately represented by non-informative distributions, giving equal likelihood to all locations and scales of the visual object in the image. This important aspect has been largely ignored in most existing work on groupwise model building (see [9; 33] for example), but has the potential to seriously impact the quality of the resulting correspondences, since the basic assumptions made in their generative formulation may be invalid.

With this formulation, the complete data likelihood in Equation (B.2) is now given by:

$$p(\mathscr{I}, \mathbf{s}, \mathbf{p}, \mathbf{q}|\boldsymbol{\theta}) = \underbrace{p(\mathscr{I}|\mathbf{s}, \mathbf{p}, \boldsymbol{\theta}_t)}_{\text{image likelihood}} \quad \underbrace{p(\mathbf{s}|\mathbf{q}, \boldsymbol{\theta}_s)}_{\text{shape likelihood}} \quad \underbrace{p(\mathbf{p})\, p(\mathbf{q})}_{\text{deformation priors}}, \tag{B.8}$$

with the deformation priors modelled as isotropic Gaussian distributions:

$$p(\mathbf{p}) = (2\pi)^{-\frac{M_t}{2}} \exp\left\{-\frac{1}{2}\|\mathbf{p}\|^2\right\} \quad \text{and} \quad p(\mathbf{q}) = (2\pi)^{-\frac{M_s}{2}} \exp\left\{-\frac{1}{2}\|\mathbf{q}\|^2\right\}. \tag{B.9}$$

The shape and image likelihoods are modelled by assuming isotropic Gaussian distributions on their respective residuals:

$$p(\mathbf{s}|\mathbf{q}, \boldsymbol{\theta}_s) = \left(2\pi\sigma_s^2\right)^{-n} \exp\left\{-\frac{1}{2\sigma_s^2}\left\|\mathbf{s} - \mathbf{R}\left(\bar{\mathbf{s}} + \boldsymbol{\Phi}_s\mathbf{q}\right) - \mathbf{T}\right\|^2\right\} \tag{B.10}$$

$$p(\mathscr{I}|\mathbf{s}, \mathbf{p}, \boldsymbol{\theta}) = \left(2\pi\sigma_t^2\right)^{-\frac{P}{2}} \exp\left\{-\frac{1}{2\sigma_t^2}\left\|\mathscr{C}\left(\mathscr{I}, \{\mathbf{x}_i\}_{i=1}^P; \mathbf{s}\right) - \alpha\left(\bar{\mathbf{t}} + \boldsymbol{\Phi}_t\mathbf{p}\right) - \beta\mathbf{1}\right\|^2\right\}, \tag{B.11}$$

where $\boldsymbol{\theta}_s = \{\sigma_s^2, \bar{\mathbf{s}}, \boldsymbol{\Phi}_s\}$ and $\boldsymbol{\theta}_t = \{\sigma_t^2, \bar{\mathbf{t}}, \boldsymbol{\Phi}_t\}$. The image likelihood is evaluated in the canonical frame, which involves cropping the image at locations defined by a warping function,

parameterised by the shape in the image frame:

$$\mathscr{C}\left(\mathscr{I}, \{\mathbf{x}_i\}_{i=1}^{P}; \mathbf{s}\right) : \Re^{2n} \rightarrow \Re^{P} = \begin{bmatrix} \mathscr{I} \circ \mathscr{W}(\mathbf{x}_1; \mathbf{s}) \\ \vdots \\ \mathscr{I} \circ \mathscr{W}(\mathbf{x}_P; \mathbf{s}) \end{bmatrix} = \mathbf{c}^{(P)}, \tag{B.12}$$

where $\{\mathbf{x}_i\}_{i=1}^{P}$ denotes the set of P locations in the canonical frame, over which the likelihood of the image $\mathscr{I}$ is evaluated.

The effect of the selected coordinate frame for model evaluation is also an issue worthy of discussion here. In the formulation given above, the likelihood of an image is evaluated in the canonical frame, where the appearance model is defined. Another possibility is to define the model in the image frame. This method is proposed in Minimum Description Length (MDL) type methods such as [33; 127]. In those works, it is argued that evaluating the model in the image frame can lead to more compact models (in the MDL sense) than those learnt by evaluating in the canonical frame. This is because, by evaluating in the canonical frame, the optimisation of the likelihood leads to deformations that avoid *difficult* parts of the image. In the extreme case, all shapes can shrink to occupy small regions in each image with *very* similar appearance over the whole set. The image likelihood is actually maximised in this configuration despite providing useless correspondences for model building. This problem is avoided in the MDL formulation since it minimises the error of image synthesis. However, this requires the method to encode the whole image, for which the distribution of the likelihood and priors do not, in general, follow that of a Gaussian distribution. As such, representing the appearance and shape as a linear object class is not well justified. Furthermore, optimisation in the non-Gaussian case is much more complicated. In the discussions that follow, it is assumed that the first image in the training set is a template for which manual annotations are available. Keeping the template's shape fixed during the estimation process biases the solution towards the template. It is expected that this will help avoid the pathological case described above.

B.2 Marginalised Maximum Likelihood Estimation

For the purpose of estimating the parameters of the densities in Equation (B.8), it is assumed that the linear expansion coefficients of both shape and appearance are also hidden variables. To summarise, the components of the MML estimation procedure are grouped are as follows:

$$\text{data}: \mathcal{D} = \{\mathscr{I}_i\}_{i=1}^{N} \tag{B.13}$$

$$\text{hidden variables}: \mathcal{V} = \{\mathbf{s}_i, \mathbf{p}_i, \mathbf{q}_i\}_{i=1}^{N} \tag{B.14}$$

$$\text{parameters}: \boldsymbol{\theta} = \left\{\{\mathbf{R}_i, \mathbf{T}_i, \alpha_i, \beta_i\}_{i=1}^{N}, \sigma_s^2, \sigma_t^2, \boldsymbol{\Phi}_s, \boldsymbol{\Phi}_t, \bar{\mathbf{s}}, \bar{\mathbf{t}}\right\}, \tag{B.15}$$

The aim of MML parameter estimation is to maximise the complete data likelihood:

$$p\left(\mathcal{D}, \mathcal{V}|\boldsymbol{\theta}\right) = \prod_{i=1}^{N} \int_{\Re^{(2n+M_s+M_t)}} p(\mathscr{I}_i|\mathbf{s}_i, \mathbf{p}_i, \boldsymbol{\theta}_t)\, p(\mathbf{s}_i|\mathbf{q}_i, \boldsymbol{\theta}_s)\, p(\mathbf{p}_i)\, p(\mathbf{q}_i)\, d\mathbf{s}_i\, d\mathbf{p}_i\, d\mathbf{q}_i. \tag{B.16}$$

As with the pairwise case presented in Chapter 3, the integral of this equation cannot be evaluated analytically, since the relation between the image and the cropping function parameters (i.e. the shape $\mathbf{s}$) is nonlinear, requiring an approximation to be made.

For images other than the template, taking a first order Taylor expansion of the cropped image in Equation (B.12), at the current shape, results in:

$$\mathscr{C}\left(\mathscr{I}, \{\mathbf{x}_i\}_{i=1}^{P}; \mathbf{s}\right) \approx \mathscr{C}\left(\mathscr{I}, \{\mathbf{x}_i\}_{i=1}^{P}; \mathbf{s}^c\right) + \mathbf{J}\ \left(\mathbf{s} - \mathbf{s}^c\right), \tag{B.17}$$

where:

$$\mathbf{J} = \nabla_{\mathbf{s}} \mathscr{C}\left(\mathscr{I}, \{\mathbf{x}_i\}_{i=1}^{P}; \mathbf{s}^c\right) \tag{B.18}$$

With this approximation, the energy term of the image likelihood can be reformulated with respect to the hidden variables as follows:

$$\left\|\mathscr{C}\left(\mathscr{I}, \{\mathbf{x}_i\}_{i=1}^{P}; \mathbf{s}\right) - \alpha\left(\bar{\mathbf{t}} + \mathbf{\Phi}_t \mathbf{p}\right) - \beta\mathbf{1}\right\|^2 \approx \mathbf{z}\mathbf{A}^T\mathbf{A}\mathbf{z} + 2\mathbf{z}^T\mathbf{A}^T\mathbf{b} + \mathbf{b}^T\mathbf{b}, \tag{B.19}$$

where:

$$\mathbf{z} = [\ \mathbf{s}\ ;\ \mathbf{p}\ ;\ \mathbf{q}\ ] \quad , \quad \mathbf{b} = \mathbf{c}^c - \mathbf{J}\mathbf{s}^c - \alpha\bar{\mathbf{t}} - \beta\mathbf{1} \quad \text{and} \quad \mathbf{A} = \begin{bmatrix}\mathbf{J} & -\alpha\mathbf{\Phi}_t & \mathbf{0}^{(P\times M_t)}\end{bmatrix}. \tag{B.20}$$

The image likelihood is then approximated as:

$$p(\mathscr{I}|\mathbf{s}, \mathbf{p}, \boldsymbol{\theta}_t) = \left(2\pi\sigma_t^2\right)^{-\frac{P}{2}} \exp\left\{-\frac{1}{2}\left[\mathbf{z}\left(\frac{1}{\sigma_t^2}\mathbf{A}^T\mathbf{A}\right)\mathbf{z} + 2\mathbf{z}^T\left(\frac{1}{\sigma_t^2}\mathbf{A}^T\mathbf{b}\right) + \frac{1}{\sigma_t^2}\mathbf{b}^T\mathbf{b}\right]\right\}. \tag{B.21}$$

Since the shape in the template image is assumed fixed, its likelihood is also given by the form in Equation (B.21), however, in this case:

$$\mathbf{z} = [\ \mathbf{p}\ ;\ \mathbf{q}\ ] \quad , \quad \mathbf{b} = \mathbf{c}^c - \alpha\bar{\mathbf{t}} - \beta\mathbf{1} \quad \text{and} \quad \mathbf{A} = \begin{bmatrix}-\alpha\mathbf{\Phi}_t & \mathbf{0}^{(P\times M_t)}\end{bmatrix}. \tag{B.22}$$

Similarly, the shape likelihood of non-template images can be written:

$$p(\mathbf{s}|\mathbf{q}, \boldsymbol{\theta}_s) = \left(2\pi\sigma_s^2\right)^{-n} \exp\left\{-\frac{1}{2}\left[\mathbf{z}\left(\frac{1}{\sigma_s^2}\mathbf{C}^T\mathbf{C}\right)\mathbf{z} + 2\mathbf{z}^T\left(\frac{1}{\sigma_s^2}\mathbf{C}^T\mathbf{d}\right) + \frac{1}{\sigma_s^2}\mathbf{d}^T\mathbf{d}\right]\right\}, \tag{B.23}$$

where:

$$\mathbf{d} = -\mathbf{R}\bar{\mathbf{s}} - \mathbf{T} \quad \text{and} \quad \mathbf{C} = \begin{bmatrix}\mathbf{I}^{(2n\times 2n)} & \mathbf{0}^{(2n\times M_t)} & -\mathbf{R}\mathbf{\Phi}_s\end{bmatrix}. \tag{B.24}$$

For the template image, the shape is known and fixed. As such, the form in Equation (B.23) still applies. However, for the template's shape, we have:

$$\mathbf{d} = \mathbf{s} - \mathbf{R}\bar{\mathbf{s}} - \mathbf{T} \quad \text{and} \quad \mathbf{C} = \begin{bmatrix}\mathbf{0}^{(2n\times M_t)} & -\mathbf{R}\mathbf{\Phi}_s\end{bmatrix}. \tag{B.25}$$

Combining the forms in Equations (B.21) and (B.23) for the image and shape likelihood,

respectively, the complete data likelihood can be written as:

$$
\begin{aligned}
& p\left(\{\mathscr{I}_i\}_{i=1}^N | \{\boldsymbol{\theta}_i\}_{i=1}^N\right) \\
& = \prod_{i=1}^N \int_{\Xi_i} p(\mathscr{I}_i, \mathbf{z}_i | \boldsymbol{\theta}_i)\, d\mathbf{z}_i \\
& \propto \left(\sigma_t^2\right)^{-\frac{PN}{2}} \left(\sigma_s^2\right)^{-nN} \exp\left\{-\frac{1}{2}\sum_{i=1}^N C_i\right\} \prod_{i=1}^N \int_{\Xi_i} \exp\left\{-\frac{1}{2}(\mathbf{z}_i - \boldsymbol{\mu}_i)^T \mathbf{H}_i (\mathbf{z}_i - \boldsymbol{\mu}_i)\right\} d\mathbf{z}_i \\
& \propto \left(\sigma_t^2\right)^{-\frac{PN}{2}} \left(\sigma_s^2\right)^{-nN} \exp\left\{-\frac{1}{2}\sum_{i=1}^N C_i\right\} \prod_{i=1}^N \det\{\mathbf{H}_i\}^{-\frac{1}{2}} \qquad \text{(B.26)}
\end{aligned}
$$

where Ξ_i denotes the domain of integration, which is given by $\Re^{M_s+M_t}$ for the template, and $\Re^{2n+M_s+M_t}$ for all others. In the equation above, for all images apart from the template, we have:

$$
\mathbf{H}_i = \frac{1}{\sigma_t^2}\mathbf{A}_i^T\mathbf{A}_i + \frac{1}{\sigma_s^2}\mathbf{C}_i^T\mathbf{C}_i + \begin{bmatrix} \mathbf{0}^{(2n\times 2n)} & \mathbf{0}^{(2n\times(M_s+M_t))} \\ \mathbf{0}^{((M_s+M_t)\times 2n)} & \mathbf{I}^{((M_s+M_t)\times(M_s+M_t))} \end{bmatrix} \qquad \text{(B.27)}
$$

$$
\boldsymbol{\mu}_i = -\mathbf{H}_i^{-1}\left(\frac{1}{\sigma_t^2}\mathbf{A}_i^T\mathbf{b}_i + \frac{1}{\sigma_s^2}\mathbf{C}_i^T\mathbf{d}_i\right) \qquad \text{(B.28)}
$$

$$
C_i = \frac{1}{\sigma_t^2}\mathbf{b}_i^T\mathbf{b}_i + \frac{1}{\sigma_s^2}\mathbf{d}_i^T\mathbf{d}_i - \boldsymbol{\mu}^T\mathbf{H}\boldsymbol{\mu}. \qquad \text{(B.29)}
$$

Similar forms can be obtained for the components pertaining to the template image.

B.3 Estimation through Expectation Maximisation

Unlike the pairwise case described in Chapter (3), finding the optimal density parameterisations for the shape and appearance by minimising Equation (B.26) is in general extremely difficult. In the pairwise case, although the corresponding error function is a complex non-linear equation, due to the low dimensionality of the problem, direct optimisation may still be tractable. In the groupwise setting, optimisation must be performed over the appearance model. Since the dimensionality of the model's appearance is generally quite large, typically in the order of tens of thousands, direct optimisation is not applicable here. Instead, following the discussion in Section 3.4.3, the EM algorithm can be utilised here also.

An outline of the EM algorithm, utilising the formulation described in the preceding sections, is presented in Algorithm 8. Similarities can bee seen between the groupwise method described here and that of existing approaches (see [9; 33; 133], for example), where the main component is the alternation between finding the correspondences and re-estimating the shape and appearance models. The difference here is that at each step, which involves a linearisation of the images, an EM procedure is utilised to find the most suitable model parameters, including the shape and appearance models. In most existing groupwise methods, steps 3 to 14 in Algorithm 8 are replaced by a PCA procedure over the current estimates of the shapes and the appearance. Since the shape and appearance models are learnt independently of each other,

Algorithm 8 Groupwise Correspondence Learning

Require: $\{\mathscr{I}_1, \mathbf{s}_1\}$ (template), $\{\mathscr{I}_i, \mathbf{s}_i\}_{i=2}^N$ (images and initial shape estimates), N_i (number of linearisations), N_{EM} (number of EM-steps), N_M (number of M-steps), M_s (number of shape modes) and M_t (number of appearance modes)

Initialise parameters $\left\{\{\mathbf{R}_i, \mathbf{T}_i, \alpha_i, \beta_i\}_{i=1}^N, \sigma_s^2, \sigma_t^2, \mathbf{\Phi}_s, \mathbf{\Phi}_t, \bar{\mathbf{s}}, \bar{\mathbf{t}}\right\}$
for $i = 1$ to N_i **do**
  Linearise all cropped images apart from the template at their current shape estimates $\{\mathbf{s}_i^c = \mathbf{s}_i\}_{i=2}^N$ {Equation (B.17)}
  for $j = 1$ to N_{EM} **do**
    E-step: Compute $\{\boldsymbol{\mu}_i, \mathbf{H}_i\}_{i=1}^N$ {Equations (B.27) and (B.28)}
    for $k = 1$ to N_M **do**
      M-step: Update global lighting parameters $\{\alpha_i, \beta_i\}_{i=2}^N$ {Equation (B.52)}
      M-step: Update appearance model $\{\bar{\mathbf{t}}, \mathbf{\Phi}_t\}$ {Equation (B.62)}
      M-step: Update image noise variance σ_t^2 {Equation (B.58)}
      M-step: Update similarity transform parameters $\{\mathbf{R}_i, \mathbf{T}_i\}_{i=2}^N$ {Equation (B.67)}
      M-step: Update shape model $\{\bar{\mathbf{s}}, \mathbf{\Phi}_s\}$ {Equation (B.83)}
      M-step: Update shape noise variance σ_s^2 {Equation (B.78)}
    end for
  end for
  Compute new estimates of shapes $\{\mathbf{s}_i = \boldsymbol{\mu}_{i(1:2n)}\}_{i=2}^N$ {Equation (B.28)}
end for
return $\{\mathbf{s}_i\}_{i=1}^N$

a *fixed* scaling factor is used to regularise the correspondence finding procedure. Finally, it should be noted that, as with other existing groupwise methods, the procedure described in this section does not afford any proof of convergence. This is because, although the EM algorithm guarantees an increase in the data log likelihood at each step, due to the linearisation of the cropped image, at each iteration of the procedure, the EM algorithm solves a different, albeit similar, problem.

The remainder of this section is dedicated to derivations of the various components of the groupwise procedure pertaining to the EM algorithm. In particular, the expectation step is described in Section B.3.1 and the maximisation step in Section B.3.2.

B.3.1 Expectation Step

The expectation step in the EM algorithm involves building the posterior density of hidden variables. Using the formulation given in the preceding sections, the hidden variables posterior is given by:

$$\begin{aligned}
&p\left(\{\mathbf{z}_i\}_{i=1}^N|\{\mathscr{I}_i,\boldsymbol{\theta}_i\}_{i=1}^N\right)\\
&=\frac{\prod_{i=1}^N p(\mathscr{I}_i,\mathbf{z}_i|\boldsymbol{\theta}_i)}{\prod_{i=1}^N p(\mathscr{I}_i|\boldsymbol{\theta}_i)}\\
&=\frac{\left(\sigma_t^2\right)^{-\frac{PN}{2}}\left(\sigma_s^2\right)^{-nN}\exp\left\{-\frac{1}{2}\sum_{i=1}^N C_i\right\}\prod_{i=1}^N\exp\left\{-\frac{1}{2}(\mathbf{z}_i-\boldsymbol{\mu}_i)^T\mathbf{H}_i(\mathbf{z}_i-\boldsymbol{\mu}_i)\right\}}{\left(\sigma_t^2\right)^{-\frac{PN}{2}}\left(\sigma_s^2\right)^{-nN}\exp\left\{-\frac{1}{2}\sum_{i=1}^N C_i\right\}\prod_{i=1}^N\det\{\mathbf{H}_i\}^{-\frac{1}{2}}}\\
&\propto\prod_{i=1}^N\det\{\mathbf{H}_i\}^{-\frac{1}{2}}\exp\left\{-\frac{1}{2}(\mathbf{z}_i-\boldsymbol{\mu}_i)^T\mathbf{H}_i(\mathbf{z}_i-\boldsymbol{\mu}_i)\right\}\\
&\propto\prod_{i=1}^N\mathcal{N}\left(\mathbf{z}_i\,;\,\boldsymbol{\mu}_i\,,\,\mathbf{H}_i^{-1}\right)
\end{aligned}\tag{B.30}$$

Notice that the resulting posterior over the hidden variables takes the form of a multivariate Gaussian density by virtue of the linearisation of the image cropping operation described in the preceding section.

B.3.2 Maximisation Step

With the posterior density over the hidden variables defined, the maximisation step of the EM algorithm involves minimising the expected negative data log likelihood:

$$\begin{aligned}
\mathscr{Q}(\boldsymbol{\theta})=&\sum_{i=1}^N E_{p(\mathbf{z}_i|\mathscr{I}_i,\boldsymbol{\theta}_i)}\left[-\ln\{p(\mathscr{I})i,\mathbf{z}_i|\boldsymbol{\theta}_i)\}\right]\\
&\propto\frac{1}{2\sigma_t^2}\left(E_1\left[\left\|\mathbf{c}_1-\alpha_1\tilde{\boldsymbol{\Phi}}_t\tilde{\mathbf{q}}_1-\beta_1\mathbf{1}\right\|^2\right]+\sum_{i=2}^N E_i\left[\left\|\mathbf{a}_i+\mathbf{J}_i\mathbf{s}_i-\alpha_i\tilde{\boldsymbol{\Phi}}_t\tilde{\mathbf{p}}_i-\beta_i\mathbf{1}\right\|^2\right]\right)+\\
&\frac{1}{2\sigma_s^2}\left(E_1\left[\left\|\mathbf{s}_1-\mathbf{R}_1\tilde{\boldsymbol{\Phi}}_s\tilde{\mathbf{q}}_1-\mathbf{T}_1\right\|^2\right]+\sum_{i=2}^N E_i\left[\left\|\mathbf{s}_i-\mathbf{R}_i\tilde{\boldsymbol{\Phi}}_s\tilde{\mathbf{q}}_i-\mathbf{T}_i\right\|^2\right]\right)+\\
&\frac{PN}{2}\ln\left\{\sigma_t^2\right\}+nN\ln\left\{\sigma_s^2\right\}+\underbrace{\frac{1}{2}\sum_{i=1}^N E_i\left[\|\mathbf{p}_i\|^2+\|\mathbf{q}_i\|^2\right]}_{\text{constant}}
\end{aligned}\tag{B.31}$$

where $\mathbf{a}_i=\mathbf{c}_i-\mathbf{J}_i\mathbf{s}_i^c$ and a compact linear model representation is used:

$$\tilde{\mathbf{p}}=[1\,;\,\mathbf{p}]\;\;,\;\;\tilde{\mathbf{q}}=[1\,;\,\mathbf{q}]\;\;,\;\;\tilde{\boldsymbol{\Phi}}_t=\begin{bmatrix}\bar{\mathbf{t}} & \boldsymbol{\Phi}_t\end{bmatrix}\;\text{ and }\;\tilde{\boldsymbol{\Phi}}_s=\begin{bmatrix}\bar{\mathbf{s}} & \boldsymbol{\Phi}_s\end{bmatrix}.\tag{B.32}$$

In the maximisation step's objective, $E_i[\mathbf{x}] = E_{p(\mathbf{z}_i|\mathscr{I}_i,\boldsymbol{\theta}_i)}[\mathbf{x}]$ denotes the expectation of $\mathbf{x}$ given the posterior density function $p(\mathbf{z}_i|\mathscr{I}_i, \boldsymbol{\theta}_i)$.

The objective function in Equation (B.31) does not afford a closed form solution for the parameters. As such, an iterative optimisation regime must be utilised, leading to the Generalised EM algorithm, when the optimisation is terminated before a global solution is reached. As an optimisation strategy, the parameters are partitioned into groups for which their optimal settings can be obtained in closed form, given all other parameters are fixed. In the following sections, the parameter groupings and their respective updates are derived. In order to afford more compact derivations, the following terms are defined:

$$E_{p(\mathbf{z}_i|\mathscr{I}_i,\boldsymbol{\theta}_i)}[\mathbf{z}_i] = \boldsymbol{\mu}_i \tag{B.33}$$

$$E_{p(\mathbf{z}_i|\mathscr{I}_i,\boldsymbol{\theta}_i)}[\tilde{\mathbf{z}}_i] = [\ 1\ ;\ \boldsymbol{\mu}_i\ ] = \tilde{\boldsymbol{\mu}}_i \tag{B.34}$$

$$E_{p(\mathbf{z}_i|\mathscr{I}_i,\boldsymbol{\theta}_i)}\left[\mathbf{z}_i\mathbf{z}_i^T\right] = \mathbf{H}_i^{-1} + \boldsymbol{\mu}_i\boldsymbol{\mu}_i^T = \boldsymbol{\phi}_i \tag{B.35}$$

$$E_{p(\mathbf{z}_i|\mathscr{I}_i,\boldsymbol{\theta}_i)}\left[\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^T\right] = \begin{bmatrix} 1 & \boldsymbol{\mu}_i^T \\ \boldsymbol{\mu}_i & \boldsymbol{\phi}_i \end{bmatrix} = \tilde{\boldsymbol{\phi}}_i, \tag{B.36}$$

where $\tilde{\mathbf{z}}_i = [\ 1\ ;\ \mathbf{z}\ ]$. Apart from these forms, the following restructuring matrices will also be used extensively:

$$\mathbf{R}_{\mathbf{s}} = \left[\mathbf{I}^{2n\times 2n} \quad \mathbf{0}^{2n\times(M_s+M_t)}\right] \tag{B.37}$$

$$\mathbf{R}_{\mathbf{p}} = \left[\mathbf{0}^{(M_t\times(2n))} \quad \mathbf{I}^{(M_t\times M_t)} \quad \mathbf{0}^{(M_t\times M_s)}\right] \tag{B.38}$$

$$\mathbf{R}_{\mathbf{q}} = \left[\mathbf{0}^{(M_s\times(2n+M_t))} \quad \mathbf{I}^{(M_s\times M_s)}\right] \tag{B.39}$$

$$\mathbf{R}_{\tilde{\mathbf{s}}} = \begin{bmatrix} 1 & \mathbf{0}^{(1\times(2n+M_s+M_t))} \\ \mathbf{0}^{(2n)} & \mathbf{R}_{\mathbf{s}} \end{bmatrix} \tag{B.40}$$

$$\mathbf{R}_{\tilde{\mathbf{p}}} = \begin{bmatrix} 1 & \mathbf{0}^{(1\times(2n+M_s+M_t))} \\ \mathbf{0}^{(M_t)} & \mathbf{R}_{\mathbf{p}} \end{bmatrix} \tag{B.41}$$

$$\mathbf{R}_{\tilde{\mathbf{q}}} = \begin{bmatrix} 1 & \mathbf{0}^{(1\times(2n+M_s+M_t))} \\ \mathbf{0}^{(M_s)} & \mathbf{R}_{\mathbf{q}} \end{bmatrix} \tag{B.42}$$

Finally, the definition $\tilde{\mathbf{s}}_i = [\ 1\ ;\ \mathbf{s}_i\ ]$ will also be used.

The Global Lighting Parameters Update

The global lighting parameters for each image $\{\alpha_i, \beta_i\}_{i=1}^N$ are independent of their counterparts in all other images. As such, the updates can be performed for each image separately. The component of the maximisation step's objective in Equation (B.31), pertaining to the global lighting parameters of the i^{th} image, is given by:

$$\mathscr{Q}_{\{\alpha_i,\beta_i\}}(\boldsymbol{\theta}) \propto E_i\left[\left\|\mathbf{a}_i + \mathbf{J}_i\mathbf{s}_i - \alpha_i\tilde{\boldsymbol{\Phi}}_t\tilde{\mathbf{p}}_i - \beta_i\mathbf{1}\right\|^2\right] \tag{B.43}$$

Letting:

$$\mathbf{u}_i = [\alpha_i\ ;\ \beta_i] \quad , \quad \mathbf{m}_i = \mathbf{a}_i - \mathbf{J}_i\mathbf{s}_i \quad \text{and} \quad \mathbf{M}_i = \left[\tilde{\boldsymbol{\Phi}}_t\tilde{\mathbf{p}}_i \quad \mathbf{1}^{(P)}\right], \tag{B.44}$$

the objective can be rewritten as:

$$\mathscr{Q}_{\{\alpha_i,\beta_i\}}(\boldsymbol{\theta}) \propto E_i\left[\|\mathbf{m}_i\|^2\right] - 2\,E_i\left[\mathbf{m}_i^T\mathbf{M}_i\right]\mathbf{u}_i + \mathbf{u}_i^T E_i\left[\mathbf{M}_i^T\mathbf{M}_i\right]\mathbf{u}_i. \tag{B.45}$$

Taking the derivative of this objective with respect to $\mathbf{u}_i$ and equating to zero, the solution for the global lighting parameters takes the form:

$$\mathbf{u}_i = \left(E_i\left[\mathbf{M}_i^T\mathbf{M}_i\right]\right)^{-1} E_i\left[\mathbf{m}_i^T\mathbf{M}_i\right] = \begin{bmatrix} c_1 & c_2 \\ c_2 & c_3 \end{bmatrix}^{-1}\begin{bmatrix} c_4 \\ c_5 \end{bmatrix} \tag{B.46}$$

Here, the components of $E_i\left[\mathbf{M}_i^T\mathbf{M}_i\right]$ and $E_i\left[\mathbf{m}_i^T\mathbf{M}_i\right]$ can be derived as follows:

$$c_1 = E_i\left[\tilde{\mathbf{p}}_i^T\tilde{\mathbf{\Phi}}_t^T\tilde{\mathbf{\Phi}}_t\tilde{\mathbf{p}}_i\right] = \text{tr}\left\{\tilde{\mathbf{\Phi}}_t^T\tilde{\mathbf{\Phi}}_t E_i\left[\tilde{\mathbf{p}}_t\tilde{\mathbf{p}}_i^T\right]\right\} = \text{tr}\left\{\tilde{\mathbf{\Phi}}_t^T\tilde{\mathbf{\Phi}}_t\mathbf{R}_{\tilde{\mathbf{p}}}\tilde{\boldsymbol{\phi}}_i\mathbf{R}_{\tilde{\mathbf{p}}}^T\right\} \tag{B.47}$$

$$c_2 = E_i\left[\mathbf{1}^T\tilde{\mathbf{\Phi}}_t\tilde{\mathbf{p}}_i\right] = \mathbf{1}^T\tilde{\mathbf{\Phi}}_t E_i\left[\tilde{\mathbf{p}}_i\right] = \mathbf{1}^T\tilde{\mathbf{\Phi}}_t\mathbf{R}_{\tilde{\mathbf{p}}}\tilde{\boldsymbol{\mu}}_i \tag{B.48}$$

$$c_3 = E_i\left[\|\mathbf{1}\|^2\right] = P \tag{B.49}$$

$$c_4 = E_i\left[\mathbf{a}_i^T\tilde{\mathbf{\Phi}}_t\tilde{\mathbf{p}}_i + \tilde{\mathbf{p}}_i^T\tilde{\mathbf{\Phi}}_t^T\mathbf{J}_i\mathbf{s}_i\right] = \mathbf{a}_i^T\tilde{\mathbf{\Phi}}_t\mathbf{R}_{\tilde{\mathbf{p}}}\tilde{\boldsymbol{\mu}}_i + \text{tr}\left\{\tilde{\mathbf{\Phi}}_t^T\tilde{\mathbf{J}}_i\mathbf{R}_{\tilde{\mathbf{s}}}\tilde{\boldsymbol{\phi}}_i\mathbf{R}_{\tilde{\mathbf{p}}}^T\right\} \tag{B.50}$$

$$c_5 = E_i\left[\mathbf{1}^T\left(\mathbf{a}_i - \mathbf{J}_i\mathbf{R}_{\mathbf{s}}\boldsymbol{\mu}_i\right)\right] = \mathbf{1}^T\left(\mathbf{a}_i - \mathbf{J}_i\mathbf{R}_{\mathbf{s}}\boldsymbol{\mu}_i\right), \tag{B.51}$$

Using the forms derived above for the global lighting updates can be written as:

$$\begin{bmatrix}\alpha_i \\ \beta_i\end{bmatrix} = \begin{bmatrix} \text{tr}\left\{\tilde{\mathbf{\Phi}}_t^T\tilde{\mathbf{\Phi}}_t\mathbf{R}_{\tilde{\mathbf{p}}}\tilde{\boldsymbol{\phi}}_i\mathbf{R}_{\tilde{\mathbf{p}}}^T\right\} & \mathbf{1}^T\tilde{\mathbf{\Phi}}_t\mathbf{R}_{\tilde{\mathbf{p}}}\tilde{\boldsymbol{\mu}}_i \\ \mathbf{1}^T\tilde{\mathbf{\Phi}}_t\mathbf{R}_{\tilde{\mathbf{p}}}\tilde{\boldsymbol{\mu}}_i & P \end{bmatrix}^{-1}\begin{bmatrix} \mathbf{a}_i^T\tilde{\mathbf{\Phi}}_t\mathbf{R}_{\tilde{\mathbf{p}}}\tilde{\boldsymbol{\mu}}_i + \text{tr}\left\{\tilde{\mathbf{\Phi}}_t^T\tilde{\mathbf{J}}_i\mathbf{R}_{\tilde{\mathbf{s}}}\tilde{\boldsymbol{\phi}}_i\tilde{\mathbf{J}}_i\mathbf{R}_{\tilde{\mathbf{p}}}^T\right\} \\ \mathbf{1}^T\left(\mathbf{a}_i + \mathbf{J}_i\mathbf{R}_{\mathbf{s}}\boldsymbol{\mu}_i\right) \end{bmatrix}. \tag{B.52}$$

Note that the global lighting parameters for the template image can be assumed fixed at $\alpha_1 = 1$ and $\beta_1 = 0$.

The Image Noise Variance Update

The component of the maximisation step's objective in Equation (B.31), pertaining to image noise, is given by:

$$\mathscr{Q}(\boldsymbol{\theta})_{\sigma_t^2} = \frac{PN}{2}\ln\left\{\sigma_t^2\right\} + \frac{C_{\sigma_t^2}}{2\sigma_t^2}, \tag{B.53}$$

where the constant $C_{\sigma_t^2}$ takes the form:

$$C_{\sigma_t^2} = E_1\left[\left\|\mathbf{c}_i - \alpha_1\tilde{\mathbf{\Phi}}_t\tilde{\mathbf{q}}_1 - \beta_1\mathbf{1}\right\|^2\right] + \sum_{i=2}^{N} E_i\left[\left\|\mathbf{a}_i + \mathbf{J}_i\mathbf{s}_i - \alpha_i\tilde{\mathbf{\Phi}}_t\tilde{\mathbf{p}}_i - \beta_i\mathbf{1}\right\|^2\right] \tag{B.54}$$

Letting:

$$\mathbf{g}_i = \begin{cases}\mathbf{c}_i - \alpha_1\bar{\mathbf{t}} - \beta_1\mathbf{1} & \text{if } i = 1 \\ \mathbf{a}_i - \alpha_i\bar{\mathbf{t}} - \beta_i\mathbf{1} & \text{otherwise}\end{cases} \quad \text{and} \quad \mathbf{\Psi}_i = \begin{cases}\begin{bmatrix}-\alpha_1\mathbf{\Phi}_t & \mathbf{0}^{(P\times M_s)}\end{bmatrix} & \text{if } i = 1 \\ \begin{bmatrix}\mathbf{J}_i & -\alpha\mathbf{\Phi}_t & \mathbf{0}^{(P\times M_s)}\end{bmatrix} & \text{otherwise}\end{cases}, \tag{B.55}$$

the constant $C_{\sigma_t^2}$ can be evaluated as follows:

$$\begin{aligned} C_{\sigma_t^2} &= \sum_{i=1}^{N} E_i \left[\|\mathbf{g}_i + \mathbf{\Psi}_i \mathbf{z}_i\|^2 \right] \\ &= \sum_{i=1}^{N} E_i \left[\|\mathbf{g}_i\|^2 \right] + 2\mathbf{g}_i^T \mathbf{\Psi}_i E_i[\mathbf{z}_i] + \text{tr} \left\{ \mathbf{\Psi}_i^T \mathbf{\Psi}_i E_i \left[\mathbf{z}_i \mathbf{z}_i^T \right] \right\} \\ &= \sum_{i=1}^{N} \|\mathbf{g}_i\|^2 + 2\mathbf{g}_i^T \mathbf{\Psi}_i \boldsymbol{\mu}_i + \text{tr} \left\{ \mathbf{\Psi}_i^T \mathbf{\Psi} \boldsymbol{\phi}_i \right\}. \end{aligned} \tag{B.56}$$

Taking the derivative of Equation (B.53) with respect to σ_t^2 results in:

$$\frac{\partial \mathcal{Q}(\boldsymbol{\theta})_{\sigma_t^2}}{\partial \sigma_t^2} = \frac{PN}{2} \frac{1}{\sigma_t^2} - \frac{C_{\sigma_t^2}}{2\sigma_t^4}. \tag{B.57}$$

Setting this derivative to zero, the update for the image noise variance, then, takes the form:

$$\sigma_t^2 = \frac{1}{PN} \sum_{i=1}^{N} \|\mathbf{g}_i\|^2 + 2\mathbf{g}_i^T \mathbf{\Psi}_i \boldsymbol{\mu}_i + \text{tr} \left\{ \mathbf{\Psi}_i^T \mathbf{\Psi}_i \boldsymbol{\phi}_i \right\}. \tag{B.58}$$

The Appearance Model Update

The component of the maximisation step's objective in Equation (B.31), pertaining to the appearance model, is given by:

$$\mathcal{Q}_{\tilde{\mathbf{\Phi}}_t}(\boldsymbol{\theta}) \propto E_1 \left[\left\| \mathbf{c}_i - \alpha_1 \tilde{\mathbf{\Phi}}_t \tilde{\mathbf{q}}_1 - \beta_1 \mathbf{1} \right\|^2 \right] + \sum_{i=2}^{N} E_i \left[\left\| \mathbf{a}_i + \mathbf{J}_i \mathbf{s}_i - \alpha_i \tilde{\mathbf{\Phi}}_t \tilde{\mathbf{p}}_i - \beta_i \mathbf{1} \right\|^2 \right] \tag{B.59}$$

Taking the derivative of this objective with respect to $\tilde{\mathbf{\Phi}}_t$:

$$\begin{aligned} \frac{\partial \mathcal{Q}_{\tilde{\mathbf{\Phi}}_t}(\boldsymbol{\theta})}{\partial \tilde{\mathbf{\Phi}}_t} &\propto E_1 \left[\alpha_1 \left(\mathbf{c}_1 - \alpha_1 \tilde{\mathbf{\Phi}}_t \tilde{\mathbf{q}}_1 - \beta_1 \mathbf{1} \right) \tilde{\mathbf{p}}_1^T \right] + \\ &\quad \sum_{i=2}^{N} E_i \left[\alpha_i \left(\mathbf{a}_i - \mathbf{J}_i \mathbf{s}_i - \alpha_i \tilde{\mathbf{\Phi}}_t \tilde{\mathbf{p}}_i - \beta_i \mathbf{1} \right) \tilde{\mathbf{p}}_i^T \right] \\ &= \alpha_1 \left(\mathbf{c}_1 - \beta_1 \mathbf{1} \right) E_i \left[\tilde{\mathbf{p}}_1^T \right] - \alpha_1^2 \tilde{\mathbf{\Phi}}_t E_i \left[\tilde{\mathbf{p}}_i \tilde{\mathbf{p}}_i^T \right] + \\ &\quad \sum_{i=2}^{N} \alpha_i \mathbf{a}_i E_i \left[\tilde{\mathbf{p}}_i^T \right] + \alpha_i \tilde{\mathbf{J}}_i E_i \left[\tilde{\mathbf{s}}_i \tilde{\mathbf{p}}_i^T \right] - \alpha_i^2 \tilde{\mathbf{\Phi}}_t E_i \left[\tilde{\mathbf{p}}_i \tilde{\mathbf{p}}_i^T \right] - \alpha_i \beta_i \mathbf{1} E_i \left[\tilde{\mathbf{p}}_i^T \right] \\ &= \alpha_1 \left(\mathbf{c}_1 - \beta_1 \mathbf{1} \right) \tilde{\boldsymbol{\mu}}_1^T \hat{\mathbf{R}}_{\tilde{\mathbf{p}}}^T - \alpha_1^t \tilde{\mathbf{\Phi}}_t \hat{\mathbf{R}}_{\tilde{\mathbf{p}}} \tilde{\boldsymbol{\phi}}_1 \hat{\mathbf{R}}_{\tilde{\mathbf{p}}}^T \\ &\quad \sum_{i=2}^{N} \alpha_i \mathbf{a}_i \tilde{\boldsymbol{\mu}}_i^T \mathbf{R}_{\tilde{\mathbf{p}}}^T + \alpha_i \tilde{\mathbf{J}}_i \mathbf{R}_{\tilde{\mathbf{s}}} \tilde{\boldsymbol{\phi}}_i \mathbf{R}_{\tilde{\mathbf{p}}}^T - \alpha_i^2 \tilde{\mathbf{\Phi}}_t \mathbf{R}_{\tilde{\mathbf{p}}} \tilde{\boldsymbol{\phi}}_i \mathbf{R}_{\tilde{\mathbf{p}}}^T - \alpha_i \beta_i \mathbf{1} \tilde{\boldsymbol{\mu}}_i^T \mathbf{R}_{\tilde{\mathbf{p}}}^T, \end{aligned} \tag{B.60}$$

where:

$$\hat{\mathbf{R}}_{\tilde{\mathbf{p}}} = \begin{bmatrix} 1 & \mathbf{0}^{(1\times M_t)} & \mathbf{0}^{(1\times M_s)} \\ \mathbf{0}^{(M_t\times 1)} & \mathbf{I}^{(M_t\times M_t)} & \mathbf{0}^{(M_t\times M_s)} \end{bmatrix} \tag{B.61}$$

Equating the derivative with zero, the appearance model updates, then, take the form:

$$\tilde{\mathbf{\Phi}}_t = \left(\alpha_1(\mathbf{c}_1 - \beta_1\mathbf{1})\tilde{\boldsymbol{\mu}}_1\hat{\mathbf{R}}_{\tilde{\mathbf{p}}}^T + \sum_{i=2}^{N}\alpha_i\left[(\mathbf{a}_i - \beta_i\mathbf{1})\tilde{\boldsymbol{\mu}}_i^T\mathbf{R}_{\tilde{\mathbf{p}}}^T + \tilde{\mathbf{J}}_i\mathbf{R}_{\tilde{\mathbf{s}}}\tilde{\boldsymbol{\phi}}_i\mathbf{R}_{\tilde{\mathbf{p}}}^T\right]\right)\times \left(\alpha_1^2\hat{\mathbf{R}}_{\tilde{\mathbf{p}}}\tilde{\boldsymbol{\phi}}_1\hat{\mathbf{R}}_{\tilde{\mathbf{p}}}^T + \sum_{i=2}^{N}\alpha_i^2\mathbf{R}_{\tilde{\mathbf{p}}}\tilde{\boldsymbol{\phi}}_i\mathbf{R}_{\tilde{\mathbf{p}}}^T\right)^{-1}. \tag{B.62}$$

The Similarity Transform Update

The similarity transform parameters for each image $\{a_i, b_i, t_{x_i}, t_{y_i}\}_{i=1}^N$ are independent of their counterparts in all other images. As such, the updates can be performed for each image separately. The component of the maximisation step's objective in Equation (B.31), pertaining to the similarity transform parameters of the i^{th} image, is given by:

$$\mathscr{Q}_{\{a_i,b_i,t_{x_i},t_{y_i}\}}(\boldsymbol{\theta}) \propto E_i\left[\left\|\mathbf{s}_i - \mathbf{R}_i\tilde{\mathbf{\Phi}}_s^T\tilde{\mathbf{q}}_i - \mathbf{T}_i\right\|^2\right]. \tag{B.63}$$

Letting:

$$\mathbf{v}_i = \begin{bmatrix} a_i \\ b_i \\ t_{x_i} \\ t_{y_i} \end{bmatrix} \quad , \quad \mathbf{x}_{ij} = \mathbf{s}_{i(2j-1:2j)} \quad \text{and} \quad \mathbf{G}_{ij} = \begin{bmatrix} \tilde{\mathbf{\Phi}}_{s(2j-1,:)}\tilde{\mathbf{q}}_i & -\tilde{\mathbf{\Phi}}_{s(2j,:)}\tilde{\mathbf{q}}_i & 1 & 0 \\ \tilde{\mathbf{\Phi}}_{s(2j,:)}\tilde{\mathbf{q}}_i & \tilde{\mathbf{\Phi}}_{s(2j-1,:)}\tilde{\mathbf{q}}_i & 0 & 1 \end{bmatrix}, \tag{B.64}$$

the objective can be written:

$$\mathscr{Q}_{\{a_i,b_i,t_{x_i},t_{y_i}\}}(\boldsymbol{\theta}) \propto \sum_{j=1}^{n} E_i\left[\|\mathbf{x}_{ij}\|^2\right] - 2\,E_i\left[\mathbf{x}_{ij}^T\mathbf{G}_{ij}\right]\mathbf{v}_i + \mathbf{v}_i^T E_i\left[\mathbf{G}_{ij}^T\mathbf{G}_{ij}\right]\mathbf{v}_i \tag{B.65}$$

Differentiating this form with respect to $\mathbf{v}_i$ and equating to zero, we get:

$$\frac{\partial\mathscr{Q}_{\{a_i,b_i,t_{x_i},t_{y_i}\}}(\boldsymbol{\theta})}{\partial\mathbf{v}_i} = \sum_{j=1}^{n}\left(E_i\left[\mathbf{G}_{ij}^T\mathbf{G}_{ij}\right]\mathbf{v}_i - 2\,E_i\left[\mathbf{G}_{ij}^T\mathbf{x}_{ij}\right]\right) = 0$$

$$\mathbf{v}_i = \left(\sum_{j=1}^{n} E_i\left[\mathbf{G}_{ij}^T\mathbf{G}_{ij}\right]\right)^{-1}\sum_{j=1}^{n} E_i\left[\mathbf{G}_{ij}^T\mathbf{x}_{ij}\right] \tag{B.66}$$

Evaluating the forms for $E_i\left[\mathbf{G}_{ij}^T\mathbf{G}_{ij}\right]$ and $E_i\left[\mathbf{G}_{ij}^T\mathbf{x}_{ij}\right]$, the updated similarity transform parameters take the form:

$$\begin{bmatrix} a_i \\ b_i \\ t_{x_i} \\ t_{y_i} \end{bmatrix} = \begin{bmatrix} c_1 & 0 & c2 & c3 \\ 0 & c1 & -c3 & c2 \\ c2 & -c3 & 1 & 0 \\ c3 & c2 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{j=1}^{n} \text{tr}\left\{\tilde{\mathbf{\Phi}}_{s(2j-1:2j,:)}^T \mathbf{R}_{\tilde{j}} \tilde{\boldsymbol{\phi}}_i \mathbf{R}_{\tilde{\mathbf{q}}}^T\right\} \\ \sum_{j=1}^{n} \text{tr}\left\{\tilde{\mathbf{\Phi}}_{s(2j-1:2j,:)}^T \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \mathbf{R}_{\tilde{j}} \tilde{\boldsymbol{\phi}}_i \mathbf{R}_{\tilde{\mathbf{q}}}^T\right\} \\ \sum_{j=1}^{n} \mathbf{R}_j \boldsymbol{\mu}_i \end{bmatrix}, \quad \text{(B.67)}$$

where:

$$c_1 = \sum_{j=1}^{n} \text{tr}\left\{\left(\tilde{\mathbf{\Phi}}_{s(2j-1,:)}^T \tilde{\mathbf{\Phi}}_{s(2j-1,:)} + \tilde{\mathbf{\Phi}}_{s(2j,:)}^T \tilde{\mathbf{\Phi}}_{s(2j,:)}\right) \mathbf{R}_{\tilde{\mathbf{q}}} \tilde{\boldsymbol{\phi}}_i \mathbf{R}_{\tilde{\mathbf{q}}}^T\right\} \quad \text{(B.68)}$$

$$\begin{bmatrix} c2 \\ c3 \end{bmatrix} = \sum_{j=1}^{n} \tilde{\mathbf{\Phi}}_{s(2j-1:2j,:)} \mathbf{R}_{\tilde{\mathbf{q}}} \tilde{\boldsymbol{\mu}}_i \quad \text{(B.69)}$$

Here:

$$\mathbf{R}_j = \begin{bmatrix} \mathbf{0}^{(2\times 2(j-1))} & \mathbf{I}^{2\times 2} & \mathbf{0}^{(2\times(2n-2j)+M_s+M_t)} \end{bmatrix} \quad \text{and} \quad \mathbf{R}_{\tilde{j}} = \begin{bmatrix} \mathbf{0}^{2\times 1} & \mathbf{R}_j \end{bmatrix} \quad \text{(B.70)}$$

Note that the similarity transform for the template shape can be assumed to be fixed at $\mathbf{R}_1 = \mathbf{I}^{(2\times 2)}$ and $\mathbf{T}_1 = \mathbf{0}^{(2)}$.

The Shape Noise Variance Update

The component of the maximisation step's objective in Equation (B.31), pertaining to shape noise, is given by:

$$\mathscr{Q}_{\sigma_s^2}(\boldsymbol{\theta}) = n \ln\left\{\sigma_s^2\right\} + \frac{C_{\sigma_s^2}}{2\sigma_s^2}, \quad \text{(B.71)}$$

where the constant $C_{\sigma_s^2}$ takes the form:

$$\sum_{i=1}^{N} E_i\left[\mathbf{R}_i^T\left(\mathbf{s}_i - \mathbf{R}_i \tilde{\mathbf{\Phi}}_s \tilde{\mathbf{q}}_i - \mathbf{T}_i\right)\tilde{\mathbf{q}}_i^T\right]. \quad \text{(B.72)}$$

Letting:

$$\mathbf{f}_i = \begin{cases} \mathbf{s}_1 - \mathbf{R}_1\bar{\mathbf{s}} - T_1 & \text{if } \; i = 1 \\ -\mathbf{R}_i\bar{\mathbf{s}} - T_i & \text{otherwise} \end{cases} \quad \text{(B.73)}$$

$$\mathbf{\Upsilon}_i = \begin{cases} \begin{bmatrix} \mathbf{0}^{(2n\times M_s)} & -\mathbf{R}_1\mathbf{\Phi}_s \end{bmatrix} & \text{if } \; i = 1 \\ \begin{bmatrix} \mathbf{I}^{(2n\times 2n)} & \mathbf{0}^{(2n\times M_s)} & -\mathbf{R}_1\mathbf{\Phi}_s \end{bmatrix} & \text{otherwise} \end{cases}, \quad \text{(B.74)}$$

the constant $C_{\sigma_s^2}$ can be evaluated as follows:

$$\begin{aligned} C_{\sigma_s^2} &= \sum_{i=1}^{N} E_i \left[\|\mathbf{f}_i + \boldsymbol{\Upsilon}_i \mathbf{z}_i\|^2 \right] \\ &= \sum_{i=1}^{N} E_i \left[\|\mathbf{f}_i\|^2 \right] + 2\mathbf{f}_i^T \boldsymbol{\Upsilon}_i E_i[\mathbf{z}_i] + \text{tr}\left\{ \boldsymbol{\Upsilon}_i^T \boldsymbol{\Upsilon}_i E_i \left[\mathbf{z}_i \mathbf{z}_i^T \right] \right\} \qquad \text{(B.75)} \\ &= \sum_{i=1}^{N} \|\mathbf{f}_i\|^2 + 2\mathbf{f}_i^T \boldsymbol{\Upsilon}_i \boldsymbol{\mu}_i + \text{tr}\left\{ \boldsymbol{\Upsilon}_i^T \boldsymbol{\Upsilon}_i \boldsymbol{\phi}_i \right\} \qquad \text{(B.76)} \end{aligned}$$

Taking the derivative of Equation (B.71) with respect to σ_s^2 results in:

$$\frac{\partial \mathscr{Q}_{\sigma_s^2}(\boldsymbol{\theta})}{\partial \sigma_s^2} = \frac{nN}{\sigma_s^2} - \frac{C_{\sigma_s^2}}{2\sigma_s^4}. \qquad \text{(B.77)}$$

Setting this derivative to zero, the updated shape noise variance, then, takes the form:

$$\sigma_s^2 = \frac{1}{2nN} \sum_{i=1}^{N} \|\mathbf{f}_i\|^2 + 2\mathbf{f}_i^T \boldsymbol{\Upsilon}_i \boldsymbol{\mu}_i + \text{tr}\left\{ \boldsymbol{\Upsilon}_i^T \boldsymbol{\Upsilon}_i \boldsymbol{\phi}_i \right\}. \qquad \text{(B.78)}$$

The Shape Model Update

The component of the maximisation step's objective in Equation (B.31), pertaining to the shape model, is given by:

$$\mathscr{Q}_{\tilde{\boldsymbol{\Phi}}_s}(\boldsymbol{\theta}) \propto E_1 \left[\left\| \mathbf{s}_1 - \mathbf{R}_1 \tilde{\boldsymbol{\Phi}}_s \tilde{\mathbf{q}}_1 - \mathbf{T}_1 \right\|^2 \right] + \sum_{i=2}^{N} E_i \left[\left\| \mathbf{s}_i - \mathbf{R}_i \tilde{\boldsymbol{\Phi}}_s \tilde{\mathbf{q}}_i - \mathbf{T}_i \right\|^2 \right] \qquad \text{(B.79)}$$

Taking the derivative of this objective with respect to $\tilde{\boldsymbol{\Phi}}_s$:

$$\begin{aligned} \frac{\partial \mathscr{Q}_{\tilde{\boldsymbol{\Phi}}_s}(\boldsymbol{\theta})}{\partial \tilde{\boldsymbol{\Phi}}_s} &\propto E_i \left[\mathbf{R}_1^T \left(\mathbf{s}_1 - \mathbf{R}_1 \tilde{\boldsymbol{\Phi}}_s \tilde{\mathbf{q}}_1 - \mathbf{T}_1 \right) \tilde{\mathbf{q}}_1^T \right] + \sum_{i=2}^{N} E_i \left[\mathbf{R}_i^T \left(\mathbf{s}_i - \mathbf{R}_i \tilde{\boldsymbol{\Phi}}_s \tilde{\mathbf{q}}_i - \mathbf{T}_i \right) \tilde{\mathbf{q}}_i^T \right] \\ &= \mathbf{R}_1^T (\mathbf{s}_1 - \mathbf{T}_1) E_i \left[\tilde{\mathbf{q}}_1 \right] - \mathbf{R}_1^T \mathbf{R}_1 \tilde{\boldsymbol{\Phi}}_s E_i \left[\tilde{\mathbf{q}}_1 \tilde{\mathbf{q}}_1^T \right] + \\ &\quad \sum_{i=2}^{N} \mathbf{R}_i^T \tilde{\mathbf{B}} E_i \left[\tilde{\mathbf{s}}_i \tilde{\mathbf{q}}_i^T \right] - \mathbf{R}_i^T \mathbf{R}_i \tilde{\boldsymbol{\Phi}}_s E_i \left[\tilde{\mathbf{q}}_i \tilde{\mathbf{q}}_i^T \right] - \mathbf{R}_i^T \mathbf{T}_i E_i \left[\tilde{\mathbf{q}}_i^T \right] \qquad \text{(B.80)} \\ &= \mathbf{R}_1^T (\mathbf{s}_1 - \mathbf{T}_1) \hat{\mathbf{R}}_{\tilde{\mathbf{q}}}^T \boldsymbol{\mu}_1^T - \mathbf{R}_1^T \mathbf{R}_1 \tilde{\boldsymbol{\Phi}}_s \hat{\mathbf{R}}_{\tilde{\mathbf{q}}} \tilde{\boldsymbol{\phi}}_1 \hat{\mathbf{R}}_{\tilde{\mathbf{q}}}^T + \\ &\quad \sum_{i=2}^{N} \mathbf{R}_i^T \tilde{\mathbf{B}} \mathbf{R}_{\tilde{\mathbf{s}}} \tilde{\boldsymbol{\phi}}_i \mathbf{R}_{\tilde{\mathbf{q}}}^T - \mathbf{R}_i^T \mathbf{R}_i \tilde{\boldsymbol{\Phi}}_s \mathbf{R}_{\tilde{\mathbf{q}}} \tilde{\boldsymbol{\phi}}_i \mathbf{R}_{\tilde{\mathbf{q}}}^T - \mathbf{R}_i^T \mathbf{T}_i \mathbf{R}_{\tilde{\mathbf{q}}}^T \tilde{\boldsymbol{\mu}}_i^T, \qquad \text{(B.81)} \end{aligned}$$

where:

$$\tilde{\mathbf{B}} = \begin{bmatrix} \mathbf{0}^{(2n \times 1)} & \mathbf{I}^{2n \times 2n} \end{bmatrix} \qquad \text{(B.82)}$$

Vectorising the derivative and equating with zero, the updated shape model, then, takes the form:

$$\text{vec}\left\{\tilde{\mathbf{\Phi}}_s\right\} = \left(\hat{\mathbf{R}}_{\tilde{\mathbf{q}}}\tilde{\boldsymbol{\phi}}_1\hat{\mathbf{R}}_{\tilde{\mathbf{q}}}^T \otimes \mathbf{R}_1^T\mathbf{R}_1 + \sum_{i=2}^{N}\mathbf{R}_{\tilde{\mathbf{q}}}\tilde{\boldsymbol{\phi}}_i\mathbf{R}_{\tilde{\mathbf{q}}}^T \otimes \mathbf{R}_i^T\mathbf{R}_i\right)^{-1} \times$$
$$\text{vec}\left\{\mathbf{R}_1^T\left(\mathbf{s}_1 - \mathbf{T}_1\right)\tilde{\boldsymbol{\mu}}_1^T\hat{\mathbf{R}}_{\tilde{\mathbf{q}}}^T + \sum_{i=2}^{N}\mathbf{R}_i^T\left(\tilde{\mathbf{B}}\mathbf{R}_{\tilde{\mathbf{s}}}\tilde{\boldsymbol{\phi}}_i\mathbf{R}_{\tilde{\mathbf{q}}}^T - \mathbf{T}_i\tilde{\boldsymbol{\mu}}_i^T\mathbf{R}_{\tilde{\mathbf{q}}}^T\right)\right\}. \quad \text{(B.83)}$$

In deriving this solution, the following Kronecker product identity was used:

$$\text{vec}\left\{\mathbf{ABC}\right\} = \left(\mathbf{C}^T \otimes \mathbf{A}\right)\text{vec}\{\mathbf{B}\}. \quad \text{(B.84)}$$

DeMoLib: Deformable Model Library

All experiments in this dissertation were implemented using the platform independent C++ library `DeMoLib`, developed as part of this doctoral work. Apart form the automatic correspondence learning method presented in Chapter 3 and the iterative-discriminative methods for AAM fitting presented in Chapter 4, the library also features classes for typical shape and appearance model building as well as implementations of a number of popular AAM fitting procedures. All results in this dissertation can be reproduced entirely using this library. The source code of the library as well documentation, generated using the Doxygen documentation system[1], can be found in the enclosed CD-ROM.

The `DeMoLib` library itself is composed of two parts: the library itself and a GUI component for manual annotation, model visualisation and fitting. All experiments in this study can be reproduced using the first component only. The third component is provided to accommodate ease of analysis of extensions. Installation instructions for all components are presented in Section C.1. A tour of the various components are presented in Sections C.2, C.3 and C.4 for the library, executables and GUI components respectively. Finally, a short tutorial is presented in Section C.5 for some common tasks on which `DeMoLib` may be used.

C.1 Installation

System Requirements

`DeMoLib` is developed to be platform independent and has been installed successfully on Microsoft Windows and Unix based platforms (including Linux and OS X). The library makes extensive use of the third party library `VXL` for all image handling and linear algebra operations. The easiest way to compile `VXL` is through the common build system `CMake`, available also for Microsoft Windows and Unix based platforms. The version of `VXL` used in this thesis is version 1.9, which can be downloaded form:

`http://vxl.sourceforge.net/`

This requires `CMake` version 2.2 or higher, which can be downloaded from:

`http://www.cmake.org/`

[1] http://www.stack.nl/ dimitri/doxygen

VXL contains a large amount of contributed components for various problems in computer vision. However, DeMoLib requires only the core component of VXL. As such, non-core components can be disabled in the CMake configuration process, to reduce compilation time.

The GUI component of DeMoLib requires the OpenCV library. Although VXL also has a GUI component called VGUI, its functionality is still rudimentary. OpenCV provides an easy to use library for the various frame displays, as well as incorporating a frame grabbing functionality for extensions in real time tracking applications. The GUI component of DeMoLib has been compiled successfully using OpenCV version 1.0, which can be downloaded form:

http://sourceforge.net/projects/opencvlibrary/

Compilation

To compile DeMoLib, we make use of the CMake build system. After extracting the library into a directory of choice, the directory will contain five sub directories: src, lib, bin, config and doc. In the src directory there is a file called CMakeLists.txt. In this file change the variables VXL_DIR to the directory where VXL is installed. To build the library's makefile, change into the config directory and execute the following command:

```
> cmake -i ../src/
```

where it is assumed that CMake is globally installed. To build the whole library, including all executables, execute make from within the same directory. The directory lib will then, contain all dynamic libraries and bin will contain all executables. Notice that there are two dynamic libraries in lib: libDeMoLib and liblibsvm. The liblibsvm is a modified version of the libsvm library [24], which is used in the constrained iterative-discriminative methods in Chapter 4.

If the GUI component of DeMoLib is desired, uncomment the line:

```
#SUBDIRS(gui)
```

in CMakeLists.txt by deleting the # character before the CMake configuration step described above. Change into the subdirectory src/gui/ and change, in the CMakeLists.txt file, the variables OPENCV_INCLUDE_DIR, OPENCV_LIB_DIR and opencv_libs to reflect the configuration of the system. Compilation then, takes the same steps as described above. The bin directory will now contains GUI executables as well.

C.2 The Library

All source files for the library component of DeMoLib can be found in the src/lib/ directory. In this section, a brief overview of the main classes is presented. It should be noted, that to avoid namespace confusion, all functions are implemented within classes, where standalone functions are defined as static functions of its class. Further information regarding their use can be found in the documentation in the doc directory or the source files themselves.

Modelling

To model LDMs, DeMoLib utilises three classes. These classes and their respective descriptions are presented in Table C.3. They make extensive use of the DeMoLib_paw and

`DeMoLib_pca` classes that implement the piecewise affine warp and PCA operation respectively. Files implementing the all these classes are in files named after the class, with the extensions `.h` and `.cxx`.

Automatic Correspondence Learning

The method for pairwise correspondence learning presented in Chapter 3 is implemented in the class `DeMoLib_pwlearn` within the files `DeMoLib_pwlearn.h` and `DeMoLib_pwlearn.cxx`.

Fitting

The various AAM fitting methods evaluated in Chapter 5 are implemented in the classes listed in Table C.2. It should be noted that all versions of the iterative-discriminative method are implemented with the option of making them background invariant as described in Section 4.5.

Miscellaneous

The classes described above make use of a number of other classes that offer a number of utilities for common operations involving LDM's. Some of these are outlined in Table C.3.

C.3 The Executables

There are a number of command line executables built using the library. Each of them are outlined in Table C.4. Information regarding their input variables can be attained by using the option `-?`. For example, to get the various options of the appearance model information executable:

```
> cam_info -?
```

Although most information regarding their use can be obtained in this way, the pairwise learning executable `pwlearn` requires a configuration file to be passed to the program. In Figure C.1, the format of this configuration is given along with an example of its entries.

C.4 The GUI

To ease the process of training, development and testing, four GUI applications are provided with `DeMoLib`. The first is the manual markup application `markup`, which allows a user to manually select a number of corresponding landmarks in a set of images. It also has a feature for automatically selecting salient landmarks, useful annotating the template image in a method for pairwise learning of correspondences. A configuration file is required for this application, where the images to be annotated are described as well as the shape files to which the correspondences will be saved. An example of this configuration file is shown in Figure C.2.

The second GUI application is `cam_visualise`, the combined appearance model visualiser. An illustration of its interface is shown in Figure C.4. It allows variations in the model's

shape, appearance (denoted here with the `textr` label), and combined appearance parameter (denoted with the `apper` label) to be synthesised by moving the sliders. How the model is presented can also be modified by toggling the display of points (by pressing the `a` key), triangulation (by pressing the `t` key) and appearance (by pressing the `a` key). Finally, the synthesised appearance can be saved to an image file by pressing the `s` key, which then requires the user to enter the image's filename in the terminal.

The third GUI application is `demo_fit`, a deformable model fitting GUI. It allows the placement of a model (similarity transform) in an image and shows the evolution of the model throughout the fitting procedure's iterations. An illustration of the interface is shown in Figure C.5. As with `cam_visualise`, `demo_fit` allows a variety of ways to visualise the model (by selecting the `0` to `7` keys for the various visualisation modes), and saving the image, with the model displayed, to a file.

The final GUI application provided with `DeMoLib` is `getbb`, an application for defining bounding boxes of visual objects in images. This application was used to generate initialisations of the pairwise method, discussed in Chapter 3. It takes as its input a configuration file defining the various parameters including the images for which a bounding box is desired. In Figure C.3, an example of this configuration file is presented. It should be noted that the `OpenCV`'s object detector can be used here by setting the `Detector` variable in this configuration file as the path containing an `OpenCV` trained detector model. Otherwise, the user can manually select the bounding box in each presented image.

C.5 A Quick Tutorial

In this section, a brief tutorial is presented to illustrate the utility of `DeMoLib`. The task that is tackled by this tutorial is that of training and use of the simultaneous inverse compositional method for use in fitting. This is illustrated using some example images and their annotations, which can be found in the `data` directory.

First, the linear models of shape and appearance must be built. These can be attained using the `train_cam` executable as follows:

```
> ./train_cam "../data/*.pts" "../data/*.pnm" ../data/Tutorial
--tDim 3 -f 0
```

Once completed, the execution of this program produces four files:

- `Tutorial-level_0.mesh`: data for the piecewise affine warping function.
- `Tutorial-level_0.pdm`: a linear shape model.
- `Tutorial-level_0.tdm`: a linear appearance model.
- `Tutorial-level_0.cam`: a combined appearance model.

To visualise the built model, execute the `cam_visualise` program as follows:

```
> ./cam_visualise ../data/Tutorial-level_0.cam -a -t -p
```

where variations in shape and appearance can be synthesised by moving the sliders pertaining to the `shape`, `textr` and `apper` labels.

To train an AAM using this model, execute the following command:

```
> ./train_aam_ic 1 ../data/Tutorial ../data/Tutorial-level_#.cam
-f 0 -v
```

which trains a simultaneous inverse compositional AAM. Data for the trained AAM is written to the file `Tutorial-level_0.aam_ic_sim`. To visualise the trained model's fitting, execute the command:

```
> ./demo_fit ../data/Tutorial-level_#.aam_ic_sim
../data/im01.pnm -f 0 -i 1
```

Select the simultaneous visualisation of shape, triangulation and appearance by pressing the `6` key. To simulate fitting, press the `f` key repeatedly until the model converges.

Table C.1: LDM Modelling Classes

Class Name	Description
`DeMoLib_pdm`	Models the shape of an LDM through a point distribution model (see Section 2.1.1). It uses a linear intrinsic parameterisation composed with a 2D similarity transform to account for global translation, scale and rotation variations.
`DeMoLib_tdm`	Models the appearance of an LDM with a linear intrinsic parameterisation with a linear lighting model (see Section 2.1.2).
`DeMoLib_cam`	Simultaneously models the shape and appearance of an LDM through a combined appearance representation (see Section 2.1.3). It contains, as public members, the `DeMoLib_pdm` and `DeMoLib_tdm` objects.

Table C.2: AAM Fitting Methods

Class Name	Files	Description
DeMoLib_demo, DeMoLib_demo_pyrd	DeMoLib_demo.h	Virtual interface class for all LDM fitting procedures. DeMLib_demo_pyrd implements a Gaussian pyramid for fitting.
DeMoLib_aam_ic_po	DeMoLib_aam_ic_po.h, DeMoLib_aam_ic_po.cxx	Implementation of the project-out inverse compositional AAM [83].
DeMoLib_aam_ic_sim	DeMoLib_aam_ic_sim.h, DeMoLib_aam_ic_sim.cxx	Implementation of the simultaneous inverse compositional AAM [4].
DeMoLib_aam_ic_norm	DeMoLib_aam_ic_norm.h, DeMoLib_aam_ic_norm.cxx	Implementation of the normalisation inverse compositional AAM [4].
DeMoLib_aam_orig	DeMoLib_aam_orig.h, DeMoLib_aam_orig.cxx	Implementation of the original fixed Jacobian and linear regression AAM [30; 43].
DeMoLib_aam_di_linear	DeMoLib_aam_di_linear.h, DeMoLib_aam_di_linear.cxx	Implementation of the linear iterative-discriminative method for the asymptotically trained (Section 4.3.1) and constrained optimisation (Section 4.3.1) methods.
DeMoLib_aam_di_haar	DeMoLib_aam_di_haar.h, DeMoLib_aam_di_haar.cxx	Implementation of the Haar-like feature based iterative-discriminative method (Section 4.3.2).
DeMoLib_aam_di_svm	DeMoLib_aam_di_svm.h, DeMoLib_aam_di_svm.cxx	Implementation of the nonlinear ν-SVR based iterative-discriminative method (Section 4.3.2).
DeMoLib_aam_di_svm_rob	DeMoLib_aam_di_svm_rob.h, DeMoLib_aam_di_svm_rob.cxx	Implementation of the robust nonlinear ν-SVR based iterative-discriminative method (Section 4.4).

Table C.3: Miscellaneous Classes

Class Name	Description
`DeMoLib_io`	Various input-output tools including reading/writing shape and triangulation files, finding corresponding shape and image filenames, and loading various objects from disk.
`DeMoLib_geo`	Implements a number of geometrical procedures such as Delaunay triangulation, 2D Procrustes alignment and 3D rotation matrices.
`DeMoLib_haar`	Implements the extended Haar-like features [72] and regressor described in Section 4.3.2.
`DeMoLib_imload`	An easy to use image set loader with options of filtering, Gaussian Pyramid reductions and background segmentation.
`DeMoLib_cclass`	The colour classifier described in Section 5.5.
`DeMoLib_sampler`	A class for sampling random AAM parameter perturbations used to generate training and test sets for experiments in Chapter 5.

Table C.4: Executables

Executable	Description
`asf2pts`	Converts the IMM `.asf` points file format to that used in `DeMoLib`, which is based on FGNet's `.pts` file format.
`cam_info`	Prints information about a trained combined appearance model to a terminal.
`pertrube_cam`	Creates a datafile containing a set of perturbed combined appearance model parameters along with their optimal settings for a set of images. Used for testing the performance of LDM fitting methods.
`pwlearn`	Perform pairwise correspondence learning over a set of images.
`res2hist`	Builds histograms of convergence accuracies from a results data file (the output of a call to `test_demo`).
`test_demo`	Test a trained LDM fitting procedure.
`train_aam_orig`	Train the original AAM [].
`train_aam_ic`	Train the various inverse compositional AAM fitting methods.
`train_aam_di_linear`	Train the various linear iterative-discriminative AAM fitting method.
`train_aam_di_haar`	Train the Haar-like feature based AAM.
`train_aam_svm`	Trains the nonlinear ν-SVR based AAM. Also includes the robust method.

```
i: 0              #index in list pertaining to template
Shape: image1.pts #shape of template
ImageDir: ./      #directory containing images
InputDir: ./      #directory to write point-files to
OutputDir: ./     #directory to write point-files to
Images: {         #filenames of image files
  image1.pnm
  ...
  image10.pnm
}
Box: {            #bounding box for each image (x1,y1,x2,y2)
  10 10 20 20
  ...
  1 2 3 4
}
Input: {          #if this is specified, box is ignored!
  image1.pts
  ...
  image10.pts
}
Output: {         #output shape files
  image1.pts
  ...
  image10.pts
}
```

Figure C.1: The pairwise learning executable configuration for the executable `pwlearn`.

```
n: 10             #Number of images to annotate
ImageDir: ./      #Directory containing images
OutputDir: ./     #Directory to save annotations in
Images {          #Image names
  image1.pnm
  ...
  image10.pnm
}
Points {          #Files to store annotations in
  image1.pts
  ...
  imageN.pts
}
```

Figure C.2: An example configuration file for the `markup` application.

```
Output: box.txt   #output file
Detector: det.xml #OpenCV trained detector object (optional)
ImageDir: ./      #directory containing images
Images: {         #filenames of image files
  image1.pnm
  ...
  image10.pnm
}
```

Figure C.3: An example configuration file for the `getbb` application.

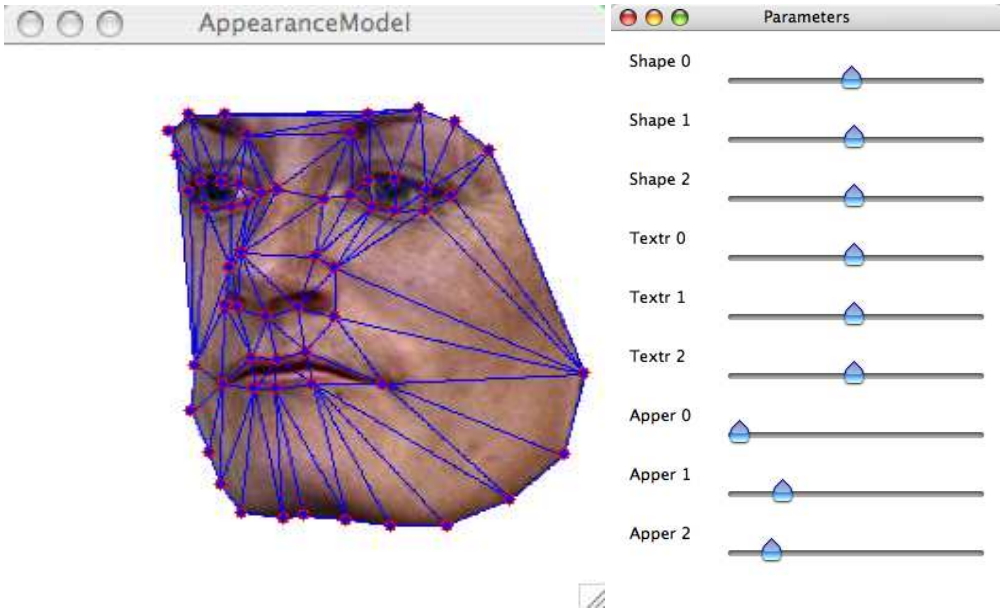


Figure C.4: The `cam_visualise` application.

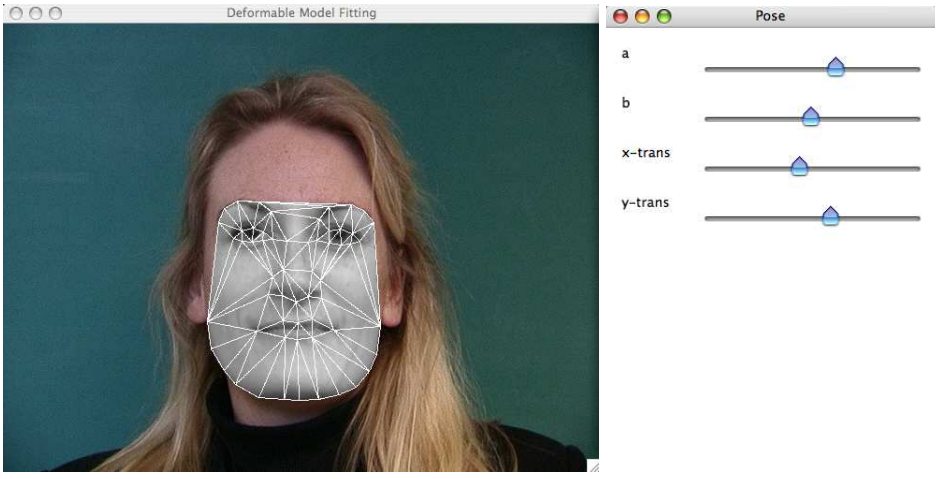


Figure C.5: The `demo_fit` application.

Bibliography

[1] T. Adamek, N. E. O'Connor, G. J. F. Jones, and N. Murphy. An Integrated Approach for Object Shape Registration and Modeling. In *MIR'05: Proceedings of the 7th ACM SIGMM International Workshop on Multimedia Information Retrieval*, 2005.

[2] K. Aström, R. Cipolla, and P. Giblin. Generalised Epipolar Constraints. *International Journal of Computer Vision (IJCV)*, 33:51–72, 1999.

[3] S. Avidan. Support Vector Tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 26:1064–1072, 2004.

[4] S. Baker, R. Gross, and I. Matthews. Lucas-Kanade 20 Years On: A Unifying Framework: Part 3. Technical report, Robotics Institute, Carnegie Mellon University, 2003.

[5] S. Baker, R. Gross, and I. Matthews. Lucas-Kanade 20 Years On: A Unifying Framework: Part 4. Technical report, Robotics Institute, Carnegie Mellon University, 2004.

[6] S. Baker, R. Gross, I. Matthews, and T. Ishikawa. Lucas-Kanade 20 Years On: A Unifying Framework: Part 2. Technical report, Robotics Institute, Carnegie Mellon University, 2003.

[7] S. Baker and I. Matthews. Equivalence and Efficiency of Image Alignment Algorithms. In *CVPR'01: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 1090–1097, 2001.

[8] S. Baker and I. Matthews. Lucas-Kanade 20 Years On: A Unifying Framework: Part 1. Technical report, Robotics Institute, Carnegie Mellon University, 2002.

[9] S. Baker, I. Matthews, and J. Schneider. Automatic Construction of Active Appearance Models as an Image Coding Problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 26:1380 – 1384, 2004.

[10] A. Bartoli, M. Perriollat, and S. Chambon. Generalized Thin-Plate Spline Warps. In *CVPR'07: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[11] C. Basso, T. Vetter, and V. Blanz. Regularized 3D Morphable Models. In *HLK '03: Proceedings of the 1st IEEE International Workshop on Higher-Level Knowledge in 3D Modeling and Motion Analysis*, page 3, 2003.

[12] A. Batur and M. Hayes. Adaptive Active Appearance Models. *IEEE Transactions on Image Processing (TIP)*, 14:1707–1721, 2005.

[13] R. Bhotika. *Scene-space Methods for Bayesian Inference of 3D shape and motion*. PhD thesis, University of Rochester Computer Science Department, 2003.

[14] C. Bishop. *Neural Networks for Pattern Recognition*. Clarendon Press, Oxford, 1995.

[15] M. Black and P. Anandan. The Robust Estimation of Multiple Motions: Affine and Piecewise-smooth Flow Fields. Technical report, Xerox PARC, 1993.

[16] A. Blake, M. Isard, and D. Reynard. Learning to Track Curves in Motion. In *IEEE International Conference on Decision Theory and Control*, pages 3788–3793, 1994.

[17] V. Blanz, C. Basso, T. Poggio, and T. Vetter. Reanimating faces in images and video. In *EUROGRAPHICS 2003*, volume 22, pages 641–650, 2003.

[18] V. Blanz and T. Vetter. A Morphable Model for the Synthesis of 3D Faces. In *SIGGRAPH'99: Proceedings of the 26th International Conference on Computer Graphics and Interactive Techniques*, pages 187–194, 1999.

[19] G. Borshukov and J. Lewis. Realistic Human Face Rendering for "The Matrix Reloaded". In *SIGGRAPH'03: Proceedings of the International Conference on Computer Graphics and Interactive Techniques Sketches & Applications*, 2003.

[20] M. Brand. Incremental Singular Value Decomposition of Uncertain Data with Missing Values. In *ECCV'02: Proceedings of the 7th European Conference on Computer Vision*, pages 707–720, 2002.

[21] M. Brand. A Direct Method for 3D Factorization of Nonrigid Motion Observed in 2D. In *CVPR'05: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 122–128, 2005.

[22] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High Accuracy Optical Flow Estimation Based on a Theory of Warping. In *ECCV'04: Proceedings of the 8th European Conference on Computer Vision*, volume 4, pages 25–36, 2004.

[23] A. Bruhn, J. Weickert, T. Kohlberger, and C. Schnoerr. A Multigrid Platform for Real-Time Motion Computation with Discontinuity-Preserving Variational Methods. *International Journal of Computer Vision (IJCV)*, 70:257–277, 2006.

[24] *LIBSVM: a Library for Support Vector Machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[25] Y. Q. Cheng, X. G. Wang, R. T. Collins, E. M. Riseman, and A. R. Hanson. Three-Dimensional Reconstruction of Points and Lines with Unknown Correspondence across Images. *International Journal of Computer Vision (IJCV)*, 45:129–156, 2001.

[26] C. M. Christoudias and T. Darrell. On Modelling Nonlinear Shape-and-Texture Appearance Manifolds. In *CVPR'05: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1067–1074, 2005.

[27] H. Chui, L. Win, R. Schultz, J. S. Duncan, and A. Rangarajan. A Unified Non-rigid Feature Registration Method for Brain Mapping. *Medical Image Analysis*, 7:113–130, 2003.

[28] T. F. Cootes, D. Cooper, C. J. Taylor, and J. Graham. Active Shape Models - Their Training and Application. *Computer Vision and Image Understanding*, 61:38–59, 1995.

[29] T. F. Cootes, G. J. Edwards, and C. J. Taylor. A Comparative Evaluation of Active Appearance Model Algorithms. In *BMVC'98: Proceedings of the 9th British Machine Vision Conference*, volume 2, pages 680–689, 1998.

[30] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active Appearance Models. In *ECCV'98: Proceedings of the 5th European Conference on Computer Vision*, volume 2, pages 484–498, 1998.

[31] T. F. Cootes and C. J. Taylor. Active Shape Models - 'Smart Snakes'. In *BMVC'92: Proceedings of the 3rd British Machine Vision Conference*, pages 266–275, 1992.

[32] T. F. Cootes and C. J. Taylor. Anatomical Statistical Models and their Role in Feature Extraction. In *British Journal of Radiology*, volume 77, pages S133–S139, 2004.

[33] T. F. Cootes, C. J. Twining, V. Petrovic, R. Schestowitz, and C. J. Taylor. Groupwise Construction of Appearance Models using Piece-wise Affine Deformations. In *BMVC'05: Proceedings of the 16th British Machine Vision Conference*, volume 2, pages 879–888, 2005.

[34] T. F. Cootes, G. V. Wheeler, K. N. Walker, and C. J. Taylor. View-Based Active Appearance Models. *Image and Vision Computing (IVC)*, 20:657–664, 2002.

[35] D. Cristinacce and T. Cootes. Boosted Active Shape Models. In *BMVC'07: Proceedings of the 18th British Machine Vision Conference*, volume 2, pages 880–889, 2007.

[36] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry*. 2nd edition. Springer-Verlag, 2000.

[37] F. De la Torre Frade, A. C. Romea, J. Cohn, and T. Kanade. Filtered Component Analysis to Increase Robustness to Local Minima in Appearance Models. In *CVPR'07: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2007.

[38] E. den Boef and D. den Hertog. Efficient Line Search Methods for Convex Functions. *SIAM Journal on Optimization*, 18:338–363, 2007.

[39] E. C. di Mauro, T. F. Cootes, G. J. Page, and C. B. Jackson. Check!: A Generic and Specific Industrial Inspection Tool. *VISP'96: Proceedings of the Conference on Vitalizing City Centres through Integrated Spatial Planning*, 143:241–249, 1996.

[40] R. Donner, M. Reiter, G. Langs, P. Peloschek, and H. Bischof. Fast active appearance model search using canonical correlation analysis. *Transactions Pattern Analysis and Machine Intelligence (TPAMI)*, 28:1690–1694, 2006.

[41] R. O. Duda and P. E. Heart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, 1973.

[42] G. J. Edwards, T. F. Cootes, and C. J. Taylor. Face Recognition using Active Appearance Models. In *ECCV'98: Proceedings of the 5th European Conference on Computer Vision*, pages 581–595, 1998.

[43] G. J. Edwards, C. J. Taylor, and T. F. Cootes. Interpreting Face Images using Active Appearance Models. In *FG'98: Proceedings of the 3rd International Conference on Automatic Face and Gesture Recognition*, pages 300–305, 1998.

[44] N. Faggian, A. P. Paplinski, and T. Chin. Face Recognition from Video using Active Appearance Model Segmentation. In *ICPR'06: Proceedings of the 18th International Conference on Pattern Recognition*, volume 1, pages 287–290, 2006.

[45] N. Faggian, S. Romdhani, J. Sherrah, and A. Paplinski. Color Active Appearance Model Analysis using a 3D Morphable Model. *DICTA'05: Proceedings of the Digital Image Computing on Techniques and Applications*, page 59, 2005.

[46] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes. *Computer Graphics: Principles and Practice*. Addison-Wesley, 1996.

[47] Y. Freund. Boosting a Weak Learning Algorithm by Majority. *Information and Computation*, 121:256–285, 1995.

[48] J. H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics*, 29:1189–1232, 2001.

[49] S. L. Gallou, G. Breton, R. Séguier, and C. Garcia. Avatar Puppetry Using Real-Time Audio and Video Analysis. In *Intelligent Virtual Agents*, pages 391–392, 2007.

[50] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis*. 2nd edition. CRC Press, 2003.

[51] A. S. Georghiades, B. P. N, and D. J. Kriegman. From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 23:643–660, 2001.

[52] C. Goodall. Procrustes Methods in the Statistical Analysis of Shape. *Journal of Royal Statistical Society B*, 53:285–339, 1991.

[53] R. Gross, I. Matthews, and S. Baker. Constructing and Fitting Active Appearance Models With Occlusion. In *Proceedings of the IEEE Workshop on Face Processing in Video*, page 72, 2004.

[54] R. Gross, I. Matthews, and S. Baker. Generic vs. person specific active appearance models. *Image and Vision Computing (IVC)*, 23:1080–1093, 2005.

[55] R. Gross, I. Matthews, and S. Baker. Active Appearance Models with Occlusion. *Image and Vision Computing (IVC)*, 24:593–604, 2006.

[56] R. Gross, L. Sweeney, F. De la Torre Frade, and S. Baker. Model-Based Face De-Identification. In *Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, pages 161–168, 2006.

[57] G. Guo, S. Li, and K. Chan. Face Recognition by Support Vector Machines. In *FG'00: Proceedings of the 4th International Conferences on Automatic Face and Gesture Recognition*, pages 196–201, 2000.

[58] G. D. Hager and P. N. Belhumeur. Efficient Region Tracking with Parametric Models of Geometry and Illumination. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, volume 20, pages 1025–1039, 1998.

[59] D. W. Hansen, J. P. Hansen, M. Niels, and M. B. Stegmann. Eye Typing using Markov and Active Appearance Models. In *WACV'02: Proceedings of the 6th IEEE Workshop on Applications of Computer Vision*, page 132, 2002.

[60] A. Hill and C. J. Taylor. A Method of Non-rigid Correspondence for Automatic Landmark Identification. In *BMVC'96: Proceedings of the 7th British Machine Vision Conference*, volume 2, pages 323–332, 1996.

[61] B. K. P. Horn and B. G. Schunck. Determining Optical Flow. *Artificial Intelligence*, 17:185–203, 1981.

[62] X. Hou, S. Li, H. Zhang, and Q. Cheng. Direct appearance models. In *CVPR'01: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 828–833, 2001.

[63] T. Jebara. Images as Bags of Pixels. In *ICCV'03: Proceedings of the 9th International Conference on Computer Vision*, volume 1, pages 265–272, 2003.

[64] T. Jebara. Kernelizing Sorting, Permutation and Alignment for Minimum Volume PCA. In *COLT'04: Proceedings of the 7th Annual Conference on Learning Theory*, pages 609–623, 2004.

[65] P. Kittipanya-ngam and T. F. Cootes. The Effect of Texture Representation on AAM Performance. In *ICPR'06: Proceedings of the 18th International Conference on Pattern Recognition*, volume 2, pages 328–331, 2006.

[66] K. Krajsek and R. Mester. Bayesian Model Selection for Optical Flow Estimation. *Pattern Recognition*, 4713:142–151, 2007.

[67] G. Langs, P. Peloschek, R. Donner, M. Reiter, and H. Bischof. Active Feature Models. In *ICPR'06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 417–420, 2006.

[68] R. Larsen, M. B. Stegmann, S. Darkner, S. Forchhammer, T. F. Cootes, and B. K. Ersbøll. Texture Enhanced Appearance Models. *Computer Vision and Image Understanding*, 106:20–30, 2007.

[69] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back. Face Recognition: A Convolutional Neural Network Approach. *IEEE Transactions on Neural Networks*, 8:98–113, 1997.

[70] C. Lee and T. Huang. Finding Point Correspondences and Determining Motion of Rigid Object from two Weak Perspective Views. *Computer Vision, Graphics and Image Processing*, 52:309–327, 1990.

[71] H. Lester and S. Arridge. A Survey of Hierarchical Non-linear Medical Image Registration. *Pattern Recognition*, 32:129–149, 1999.

[72] R. Lienhart and J. Maydt. An Extended Set of Haar-like Features for Rapid Object Detection. In *ICIP'02: Proceedings of the International Conference on Image Processing*, volume 1, pages 900–903, 2002.

[73] Y. Lin and D. Lee. Bayesian l1-norm sparse learning. In *ICASSP'06: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 5, 2006.

[74] D. C. Liu and J. Nocedal. On the Limited Memory Method for Large Scale Optimization. *Mathematical Programming B*, 45:503–528, 1989.

[75] X. Liu. Generic Face Alignment using Boosted Appearance Model. In *CVPR'07: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[76] B. D. Lucas and T. Kanade. An Iterative Image Registration Technique with an Application Stereo Vision. In *IJCAI'81: Proceedings of the 7th International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[77] N. Magnenat-Thalmann and D. Thalmann. Deformable Avatars. In *DEFORM'00: Proceedings of the Workshop on Image Registration in Deformable Environments*, volume 196, pages 29–30, 2000.

[78] P. C. Mahalanobis. On the Generalized Distance in Statistics. In *National Institute of Science of India 12*, pages 49–55, 1936.

[79] S. Mallat and Z. Zhang. Matching Pursuit with Time-frequency Dictionaries. *IEEE Transactions on Signal Processing*, 41:3397–3415, 1993.

[80] A. Martinez and R. Benavente. The AR Face Database. Technical report, CVC, 1998.

[81] L. Mason, J. Baxter, P. Bartlett, and M. Frean. Boosting Algorithms as Gradient Descent. *Advances in Neural Information Processing Systems (NIPS)*, 12:512–518, 2000.

[82] I. Matthews and S. Baker. Active Appearance Models Revisited. Technical report, Robotics Institute, Carnegie Mellon University, 2003.

[83] I. Matthews and S. Baker. Active Appearance Models Revisited. *International Journal of Computer Vision (IJCV)*, 60:135–164, 2004.

[84] R. Meir. Empirical Risk Minimization Versus Maximum Likelihood Estimation: A Case Study. *Neural Computation*, 7(1):144–157, 1995.

[85] K. Messer, J. Matas, J. Kittler, J. Lüttin, and G. Maitre. XM2VTSDB: The Extended M2VTS Database. In *AVBPA'99: International Conference on Audio- and Video-Based Biometric Person Authentication*, pages 72–77, 1999.

[86] C. D. Meyer. *Matrix Analysis and Applied Linear Algebra*, chapter 7. Society for Industrial and Applied Mathematics, 2000.

[87] P. Mittrapiyanuruk, G. N. DeSouza, and A. C. Kak. Accurate 3D Tracking of Rigid Objects with Occlusion Using Active Appearance Models. In *WACV-MOTION'05: Proceedings of the IEEE Workshop on Motion and Video Computing*, volume 2, pages 90–95, 2005.

[88] C. Neti, G. Potaminos, J. Luettin, I. Matthews, H. Glotin, and D. Vergyri. Large-Vocabulary Audio-Visual Speech Recognition: A Summary of the John Hopkins Summer 2000 Workshop. In *MMSP'01: Proceedings of the 4th IEEE Workshop on Multimedia Signal Processing*, 2001.

[89] M. M. Nordstrøm, M. Larsen, J. Sierakowski, and M. B. Stegmann. The IMM Face Database - An Annotated Dataset of 240 Face Images. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2004.

[90] R. Oliveira, J. Xavier, and J. P. Costeira. Multi-view Correspondence by Enforcement of Rigidity Constraints. *Image Vision Computing (ICV)*, 25:1008–1020, 2007.

[91] J. Peyras, A. Bartoli, H. Mercier, and P. Dalle. Segmented AAMs Improve Person-Independent Face Fitting. In *BMVC'07 - Proceedings of the 18th British Machine Vision Conference*, 2007.

[92] J. C. Platt. Fast Training of Support Vector Machines Using Sequential Minimal Optimization. *Advances in Kernel Methods: Support Vector Learning*, pages 185–208, 1999.

[93] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. Mutual-Information-based Registration of Medical Images: a Survey. *IEEE Transactions on Medical Imaging*, 22:986–1004, 2003.

[94] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1992.

[95] A. Rahimi, B. Recht, and T. Darrell. Learning Appearance Manifolds from Video. In *CVPR'05: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 868–875, 2005.

[96] B. Rasolzadeh, L. Petersson, and N. Pettersson. Response Binning: Improved Weak Classifiers for Boosting. In *IEEE Intelligent Vehicles Symposium*, pages 344–349, 2006.

[97] B. Reinhard, B. Horst, L. Franz, and M. Sonka. Robust Active Appearance Models and their Application to Medical Image Analysis. *IEEE Transactions on Medical Imaging*, 24:1151–1169, 2005.

[98] M. Roberts, T. Cootes, and J. Adams. Vertebral Morphometry: Semi-automatic Determination of Detailed Vertebral Shape from DXA Images using Active Appearance Models. *Investigative Radiology*, 41:849–859, 2007.

[99] M. G. Roberts, T. F. Cootes, and J. E. Adams. Robust Active Appearance Models with Iteratively Rescaled Kernels. In *BMVC'07: Proceedings of the 18th British Machine Vision Conference*, volume 1, pages 302–311, 2007.

[100] A. Roche, G. Malandain, X. Pennec, and N. Ayache. The Correlation Ratio as a New Similarity Measure for Multimodal Image Registration. In *MICCAI'98: Proceedings of the 1st International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 1115–1124, 1998.

[101] S. Romdhani. *Face Image Analysis using a Multiple Feature Fitting Strategy*. PhD thesis, University of Basel, Switzerland, 2005.

[102] S. Romdhani, V. Blanz, and T. Vetter. Face Identification by Fitting a 3D Morphable Model using Linear Shape and Texture Error Functions. In *ECCV'02: Proceedings of the 7th European Conference on Computer Vision*, volume 4, pages 3–19, 2002.

[103] S. Romdhani, S. Gong, and A. Psarrou. A Multi-view Nonlinear Active Shape Model using Kernel PCA. In *BMVC'99: Proceedings of the 10th British Machine Vision Conference*, pages 438–492, 1999.

[104] S. Romdhani and T. Vetter. Efficient, Robust and Accurate Fitting of a 3D Morphable Model. In *ICCV'03: Proceedings of the 9th International Conference on Computer Vision*, volume 1, pages 59–66, 2003.

[105] S. Romdhani and T. Vetter. Estimating 3D Shape and Texture Using Pixel Intensity, Edges, Specular Highlights, Texture Constraints and a Prior. In *CVPR'05: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 986–993, 2005.

[106] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear Total Variation Based Noise Removal Algorithms. *Physica D*, 60:259–268, 1992.

[107] B. Schölkopf, S. Mika, A. Smola, G. Ratsch, and K. Muller. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, 10:1299–1319, 1998.

[108] B. Schölkopf, A. Smola, R. Williamson, and P. L. Bartlett. New Support Vector Algorithms. *Neural Computation*, 12:1207–1245, 1998.

[109] A. Shashua. Trilinear Tensor: The Fundamental Construct of Multiple-view Geometry and Its Applications. In *AFPAC '97: Proceedings of the International Workshop on Algebraic Frames for the Perception-Action Cycle*, pages 190–206, 1997.

[110] N. Slesareva, A. Bruhn, and J. Weickert. Optic Flow Goes Stereo: A Variational Method for Estimating Discontinuity-preserving Dense Disparity Maps. In *Lecture Notes in Computer Science*, volume 3663, pages 33–40. Springer, 2005.

[111] L. H. Staib and J. S. Duncan. Boundary Finding with Parameterically Deformable Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 14:1061–1075, 1992.

[112] M. B. Stegmann. An Annotated Dataset of 14 Cardiac MR Images. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, April 2002.

[113] M. B. Stegmann. The AAM-API: An Open Source Active Appearance Model Implementation. In *MICCAI'03: Proceedings of the 6th International Conference on Medical Image Computing and Computer Assisted Intervention*, pages 951–952, 2003.

[114] M. B. Stegmann, B. K. Ersbøll, and R. Larsen. FAME - A Flexible Appearance Modelling Environment. *IEEE Transactions on Medical Imaging*, 22:1319–1331, 2003.

[115] M. B. Stegmann, K. Sjöstrand, and R. Larsen. Sparse Modeling of Landmark and Texture Variability using the Orthomax Criterion. In *International Symposium on Medical Imaging 2006*, volume 6144, 2006.

[116] A. Suinesiaputra, A. F. Frangi, M. Üzümcü, J. H. C. Reiber, and B. P. F. Lelieveldt. Extraction of Myocardial Contractility Patterns from Short-axes MR Images Using Independent Component Analysis. In *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis*, pages 75–86, 2004.

[117] A. Suinesiaputra, M. Üzümcü, A. F. Frangi, T. A. M. Kaandorp, J. H. C. Reiber, and B. P. F. Lelieveldt. Detecting Regional Abnormal Cardiac Contraction in Short-axis MR Images using Independent Component Analysis. In *MICCAI'04: Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention*, volume 3216, pages 737–744, 2004.

[118] J. Sung and D. Kim. A Background Robust Active Appearance Model using Active Contour Technique. *Pattern Recognition*, 40:108–120, 2007.

[119] B. Theobald, I. Matthews, and S. Baker. Evaluating Error Functions for Robust Active Appearance Models. In *FG'06: Proceedings of the 7th International Conferences on Automatic Face and Gesture Recognition*, pages 149–154, 2006.

[120] B. Theobald, G. C. S. Kruse, and J. A. Bangham. Towards a Low Bandwidth Talking Head Using Appearance Models. In *Journal of Image and Vision Computing (IVC)*, volume 21, pages 1077–1205, 2003.

[121] B. Theobald and N. Wilkinson. Real-time Speech Driven Talking Heads using Active Appearance Models. In *AVSP'07: Proceedings of the International Conference on Auditory-Visual Speech Processing*, 2007.

[122] J. P. Thirion. Image matching as a diffusion process: An analogy with Maxwell's demons. *Medical Image Analysis*, 2:243–260, 1998.

[123] M. Tipping. The Relevance Vector Machine. In M. Kaufmann, editor, *Advances in Neural Information Processing Systems (NIPS)*, 2000.

[124] M. E. Tipping. Sparse Bayesian Learning and the Relevance Vector Machine. *Journal of Machine Learning Research*, 1:211–244, 2001.

[125] M. E. Tipping and C. M. Bishop. Probabilistic Principle Component Analysis. *Journal of Royal Statistical Society*, B:611–622, 1999.

[126] L. Torresani, A. Hertzmann, and C. Bregler. Learning Non-Rigid 3D Shape from 2D Motion. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems (NIPS) 16*. MIT Press, Cambridge, MA, 2003.

[127] C. J. Twining, T. F. Cootes, S. Marsland, V. Petrovic, R. Schestowitz, and C.J.Taylor. A Unified Information-Theoretic Approach to Groupwise Non-rigid Registration and Model Building. In *Information Processing in Medical Imaging (IPMI)*, pages 1–14, 2005.

[128] C. J. Twining, T. F. Cootes, S. Marsland, V. S. Petrovic, R. S. Schestowitz, and C. J. Taylor. Information-Theoretic Unification of Groupwise Non-Rigid Registration and Model Building. In *Medical Image Understanding and Analysis*, volume 2, pages 226–230, 2006.

[129] C. J. Twining and C. J. Taylor. Kernel Principal Component Analysis and the Construction of Non-linear Active Shape Models. In *BMVC'01: Proceedings of the 12th British Machine Vision Conference*, volume 1, pages 23–32, 2001.

[130] M. Üzümcü, A. F. Frangi, J. H. C. Reiber, and B. P. F. Lelieveldt. Independent Component Analysis in Statistical Shape Models. In *SPIE'03: Proceedings of Society of Photo-Optical Instrumentation Engineers Conference*, volume 5032, pages 375–383, 2003.

[131] M. Üzümcü, A. F. Frangi, M. Sonka, J. H. C. Reiber, and B. P. F. Lelieveldt. ICA vs. PCA Active Appearance Models: Application to Cardiac MR Segmentation. In *MICCAI'03: Proceedings of the 6th International Conference on Medical Image Computing and Computer Assisted Intervention*, volume 2878, pages 451–458, 2003.

[132] B. v. Ginneken, M. B. Stegmann, and M. Loog. Segmentation of Anatomical Structures in Chest Radiographs using Supervised Methods: A Comparative Study on a Public Database. *Medical Image Analysis*, 10:19–40, 2006.

[133] T. Vetter, M. J. Jones, and T. Poggio. A bootstrapping Algorithm for Learning Linear Models of Object Classes. In *CVPR'97: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, page 40, 1997.

[134] P. Vincent and Y. Bengio. Kernel Matching Pursuit. *Machine Learning*, 48:165–187, 2002.

[135] P. Viola and M. Jones. Rapid Object Detection Using a Boosted Cascade of Simple Features. In *CVPR'01: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 511–518, 2001.

[136] D. Vlasic, M. Brand, H. Pfister, and J. Popovic. Face Transfer with Multilinear Models. In *ACM Transactions on Graphics*, volume 24, pages 426–433, 2005.

[137] K. N. Walker, T. F. Cootes, and C. J. Taylor. Automatically Building Appearance Models from Image Sequences Using Salient Features. *Image and Vision Computing*, 20:435–440, 2002.

[138] S. Wang, Y. Wang, and B. Li. Face Decorating System Based on Improved Active Shape Models. In *ACE '06: Proceedings of the ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, page 65, 2006.

[139] Y. Wang, S. Lucey, and J. Cohn. Non-Rigid Object Alignment with a Mismatch Template Based on Exhaustive Local Search. In *NTRL'07: Proceedings of the IEEE Workshop on Non-rigid Registration and Tracking*, 2007.

[140] X. Wei, L. Yin, Z. Zhu, and Q. Ji. Avatar-mediated Face Tracking and Lip Reading for Human Computer Interaction. In *MULTIMEDIA'04: Proceedings of the 12th Annual International Conference on Multimedia*, pages 500–503, 2004.

[141] O. Williams, A. Blake, and R. Cipolla. A Sparse Probabilistic Learning Algorithm for Real-time Tracking. In *ICCV'03: Proceedings of the 9th International Conference on Computer Vision*, volume 1, pages 353–360, 2003.

[142] J. Xiao, B. Georgescu, X. Zhou, D. Comaniciu, and T. Kanade. Simultaneous Registration and Modeling of Deformable Shapes. In *CVPR'06: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2429 – 2436, 2006.

[143] A. L. Yuille, D. S. Cohen, and P. Hallinan. Feature Extraction from Faces using Deformable Templates. *International Journal of Computer Vision (IJCV)*, 8:99–112, 1992.

[144] J. Zhang, S. Zhou, L. McMillan, and D. Comaniciu. Joint Real-time Object Detection and Pose Estimation using Probabilistic Boosting Network. In *CVPR'07: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[145] L. Zhang and S. M. Seitz. Estimating Optimal Parameters for MRF Stereo from a Single Image Pair. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 29:331–342, 2007.

[146] S. Zhou and D. Comaniciu. Shape Regression Machine. In *Information Processing in Medical Imaging (IPMI)*, pages 13–25, 2007.

[147] S. Zhou, B. Georgescu, X. S. Zhou, and D. Comaniciu. Image Based Regression Using Boosting Method. In *ICCV'05: Proceedings of the 10th International Conference on Computer Vision*, volume 1, pages 541–548, 2005.

[148] S. Zhou, F. Guo, J. H. Park, G. Carneiro, J. Jackson, M. Brendel, C. Simopoulos, J. Otsuki, and D. Comaniciu. A Probabilistic, Hierarchical, and Discriminant (PHD) Framework for Rapid and Accurate Detection of Deformable Anatomic Structure. In *ICCV'07: Proceedings of the 11th International Conference on Computer Vision*, 2007.

[149] S. Zhou, J. Shao, B. Georgescu, and D. Comaniciu. Pairwise Active Appearance Model and its Application to Echocardiography Tracking. In *MICCAI'06: Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention*, volume 4190, pages 736–743, 2006.

[150] S. K. Zhou, B. Georgescu, X. S. Zhou, and D. Comaniciu. Image Based Regression Using Boosting Method. In *ICCV'05: Proceedings of the 10th International Conference on Computer Vision*, volume 1, pages 541–548, 2005.

[151] Z. Zhou, R. M. Leahy, and J. Qi. Approximate Maximum Likelihood Hyperparameter Estimation for Gibbs Priors. *IEEE Transactions on Image Processing (TIP)*, 6:844–861, 1997.

[152] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and Vision Computing (IVC)*, 21:977–1000, 2003.